
LOAD
REVISTA PARA USUARIOS DE
MSX
AÑO 1 Nº 1 ₳ 2.00 REP. ARGENTINA
a Revolución
iene
e Oriente
oftware:
Comercial
Educativo
Entretenimiento
Radiografía
e la Talent
PC-200
enguajes:
Basic
Logo
LPC
a poderosa
isketera
alent DPF-550
Talent
DPC-200

# ¡YA!
## CLUB DE USUARIOS
## Talent MSX
## Cabildo 2027 - 1° piso Capital

dialogo - D.P.

A la Talent MSX nada le es imposible

**El Club de Usuarios de MSX**
**Ya funciona en su nueva dirección: Cabildo 2027 - 1° piso Capital**

Invitamos a los felices usuarios de la TALENT MSX al **curso gratuito** de introducción al fabuloso mundo de MSX.

Participe del Club de Usuarios de MSX y encuéntrese con sus amigos que también tienen la TALENT MSX, e intercambiará programas, datos y chimentos.

Podrá probar todos los accesorios de la línea MSX, ¡¡desde disketteras hasta robots!!

Podrá ver y leer todo lo que le interese sobre la norma MSX: catálogos, libros y revistas de todo el mundo. Todo con la seguridad, respaldo y seriedad que sólo TALENT puede brindarle.

*¡Para inscribirse, no olvide traer su factura de compra!*

## Club Talent MSX

MSX es marca registrada de MICROSOFT CORPORATION.




AÑO 1 Nº 1

**Director General**
Ernesto del Castillo

**Director Editorial**
Cristian Pusso

**Director Periodístico**
Fernando Flores

**Director Financiero**
Javier Campos Malbrán

**Arte y Diagramación**
Fernando Amengual
Tamara Migelson

**Coordinador**
Ariel Testori

**Redacción**
Eduardo Mombello
Andrea Sabin Paz

**Departamento de Avisos**
Oscar Devoto

**Departamento de Publicidad**
Guillermo González Aldalur

Load Revista para usuarios de MSX es una publicación mensual editada por Editorial PROEDI S.A., Paraná 720, 5º Piso, (1017) Buenos Aires. Tel.: 46-2886 y 49-7130. Registro Nacional de la Propiedad Intelectual: E.T. M. Registrada. Queda hecho el depósito que indica la Ley 11.723 de Propiedad Intelectual.
Todos los derechos reservados.
Precio de este ejemplar: A 2
Impresión: Calcotam. Fotocromo tapa: Columbia. Fotocomposición: Van Waveren.
Los ejemplares atrasados se venderán al precio del último número en circulación.
Prohibida la reproducción total o parcial de los materiales publicados, por cualquier medio de reproducción gráfico, auditivo o mecánico, sin autorización expresa de los editores. Las menciones de modelo, marcas y especificaciones se realizan con fines informativos y técnicos, sin cargo alguno para para las empresas que los comercializan y/o los representan. Al ser informativa su misión, la revista no se responsabiliza por cualquier problema que pueda plantear la fabricación, el funcionamiento y/o la aplicación de los sistemas y los dispositivos descriptos. La responsabilidad de los artículos firmados corresponde exclusivamente a sus autores.
Distribuidor en Capital: Martino, Juan de Garay 358, P.B. Capital.
Distribuidor interior: DGP: Hipólito Yrigoyen 1450, Capital Federal. T.E. 38-9266/9800.

# Sumario

# La revolución viene de Oriente

**Desde hace tiempo el mundo occidental mira hacia el Oriente, esperando que desde allí irrumpa otra revolución tecnológica, en el campo de la computación.**
**Los japoneses, después de haber deslumbrado a todos con sus productos de la más alta calidad y bajo precio (desde relojes, cámaras fotográficas y equipos de audio hasta automóviles y robots) penetran los mercados con las más avanzadas microcomputadoras, mientras desarrollan las máquinas pensantes de la quinta generación.**
**Uno de los hallazgos que más asombraron es el standard MSX que nació en el "imperio del Sol naciente" (con las principales fábricas a la cabeza) y se extendió a Europa y los Estados Unidos, para llegar a la Argentina, con el nuevo Talent MSX. Aquí sus usuarios ya pueden apreciar las ventajas a nivel de desarrollos en hardware y en software, ya que no se trabaja para una marca particular sino para un standard universal.**
**Una verdadera transformación, que nuestra revista quiere difundir para hacer conocer los múltiples beneficios que trae.**
**LOS EDITORES**



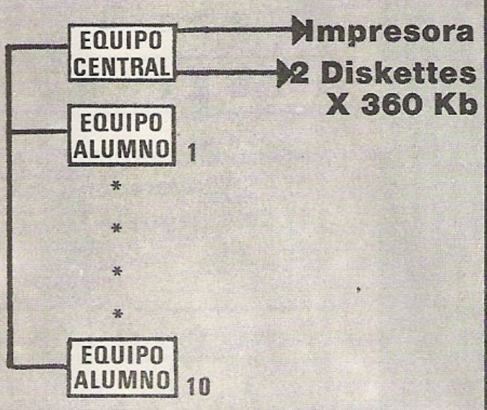

# NUEVO MODEM

La utilización de los accesos en línea a distintos servicios mediante canales standard de comunicaciones, se ha transformado en esencial para muchos tipos de actividades.
Estos desarrollos demandan dispositivos de precisión y alta velocidad, que utilicen eficientemente sus recursos.
Es por ello que TALENT ha desarrollado, con tecnología propia el MODEM-MSX, con posibilidad de incorporar un acoplador acústico o conexión directa a línea telefónica, permitiendo así la transferencia de datos entre Computadoras Personales, Bancos de Datos Públicos y Privados, Computadoras Centrales o Computadoras Profesionales.
El rango de aplicaciones del MODEM-MSX, que será lanzado al mercado en junio, incluye:
- Servicios de hoteles y pasajes (consultas de horarios y precios, disponibilidad de espacios y reservas).
- Servicio de banca hogareña (consultas de movimientos y saldos por cuenta, informes sobre transferencias entre cuentas, comercio exterior, tarjetas de crédito, información de mercado financiero, cambiario y bursátil).
- Consultas a Bancos de Datos Públicos o Privados.
- Correo Electrónico.
- Servicios de emisión y recepción de órdenes de compra o venta, etc.

El MODEM-MSX incluye además en memoria ROM, los siguientes programas incorporados, en forma standard, para:
**- Seteo del Modem:** para seleccionar protocolo BELL o CCITT, velocidades de 300 a 1200 baudios, modo de emisión o recepción.
**- Planilla Electrónica:** facilitando así una resolución rápida y sencilla de todas sus operaciones de cálculo.
**- Procesamiento de la Palabra:** completando así las funciones de manejo de textos para uso local o su incorporación al Correo Electrónico.
El Banco del Buen Ayre lo va a aplicar en el Servicio de "banca hogareña", implementándolo a partir del mes de junio. La entidad compró 1000 consolas Talent MSX con Modems que los venderá a los usuarios para que, desde sus casas, puedan averiguar los estados de cuenta corriente, caja de ahorro, plazos fijos, tarjetas de crédito y hagan pedidos de chequera, saldos, etc. En un futuro también podrán conectarse a bancos de datos. El nuevo sistema se presenta en Infocom '86, la Primera Exposición Internacional de Equipamientos, Técnicas y Servicios para la Informática, Teleinformática, Telecomunicaciones y la Oficina (Buenos Aires Sheraton Hotel del 19 al 25 de mayo). Allí, mediante dos consolas conectadas por líneas telefónicas con el banco y cualquier cliente puede consultar las cuentas y el estado de las tarjetas de crédito.

# RED LOCAL EDUCACIONAL

Telemática ofrece la Red Local Educacional, Mini-Lan, que permite que hasta 10 consolas compartan los recursos de diskette e impresora, administrados por el profesor. Sus características son:




**Operaciones previstas:**
1) Carga de un programa de la estación maestra, a todas las estaciones de alumno.
2) Carga individual de un programa de la estación maestra a una estación alumno.
3) Salvado de un programa de alumno en la maestra.
4) Salida por impresora de listado del programa alumno (vía spooling).
5) Salida de datos de alumno a display maestra, impresora o diskette.

Cada alumno puede opcionalmente usar impresora o grabador locales.
**Configuración:** Cada estación alumno usa un cartucho con el programa de MINI-LAN con la identificación correspondiente.
La estación maestra corre un programa principal para control de la red. Cuando no está atendiendo la red puede correr cualquier programa.
**Conexión física:** Se realiza mediante dos cables blindados de tipo audio.
-Velocidad de comunicaciones: 4800/9600 bit/seg.
-Costo adicional por estación alumno: ₳ 100.-
-Costo adicional por estación maestra: ₳ 50.-

# APLICACIONES COMERCIALES

Software comercial para la Talent MSX fue desarrollado por Informática San Martín y será distribuido por Prosoft S.R.L. El mismo consta de:

**Sueldos y jornales:**
El sistema permite liquidar hasta 5 empresas por diskette, lo que lo hace recomendable para estudios que realizan liquidaciones para terceros.
Quien tiene un equipo con un solo drive de diskette podrá liquidar hasta 80 empleados, por ejemplo, si usted, liquida una empresa con 30 empleados le queda espacio para 50 empleados más que pueden estar repartidos en otras cuatro empresas.
Si en cambio la MSX TALENT posee la cabina adicional, la capacidad del sistema se incrementa a 400 empleados y permite liquidar hasta 10 empresas.
El sistema emitirá "TODOS" los listados y planillas requeridos y le permitirá liquidar prácticamente todos los casos que se puedan presentar mediante la creación de nuevos conceptos y formas diversas de cálculo.
La velocidad del sistema es de 2.5 segundos para calcular la liquidación de cada empleado, en tanto que la impresión de un recibo demora alrededor de 25 segundos, para un impresor de 100 caracteres por segundo.
Para permitir la corrección de posibles errores el usuario puede emitir los listados cuantas veces resulte necesario (TODOS los listados).
Una vez que todos los procesos fueron realizados recién se procede a la grabación de la presente liquidación.
Los listados emitidos por el sistema son:
- Libro de Sueldos y Jornales.
- Recibo de sueldos.
- Planilla de haberes netos y cambios. (El sistema permite incorporar nuevos billetes).
- Totales de conceptos y aportes.
- Totales de salarios familiares.
- Obras sociales por empleado.
- Sindicato por empleado.
- U44.

**Contabilidad general:**
El sistema permite procesar hasta 5 empresas si usted tiene una sola cabina y hasta 10 con dos.
Emite los listados tradicionales, es decir: Balance de sumas y saldos, Balance Patrimonial, Diario General, Mayor analítico y Plan de Cuentas.
En este sistema llamamos CUENTA a todos los renglones integrantes del Plan de Cuentas, de este modo el Activo, el Pasivo, etc., son denominados Cuentas sin por ello alterar la lógica de los procedimientos normales.

La capacidad del sistema está definida aproximadamente por la siguiente fórmula: 187 x cant. cuentas + 32 x pases + 31 x asientos = x

El resultado de esta fórmula (x) no debe exceder 200.000.

Como ejemplo veamos el resultado del espacio ocupado por una empresa con 200 cuentas, 1000 pases en el mes que componen unos 300 asientos:

(200 x 187) + (32 x 1000) + (300 x 31) = 70330

Esto significa que usted puede tener tres empresas similares a ésta, 70330 x 3 es aproximadamente 200.000 ó bien 5 más pequeñas. En definitiva, es la suma de las cantidades de cuentas, pases y asientos totales.

## Gestión de ventas:

Características del sistema:

1. Emite las facturas, Notas de Débito y Notas de Crédito (con o sin movimiento de stock).
2. Realiza el control de stock (la factura descuenta del mismo).
3. Lleva la cuenta corriente de los clientes como saldo anterior más movimientos igual saldo actual.
4. Permite la emisión del Libro de IVA ventas y el subdiario de cuenta corriente deudores.
5. Ventas por vendedor ayudando así a la liquidación de comisiones.
6. Ingreso de pagos de clientes y de mercaderías a stock.
7. Consulta de cuenta corriente y de stock por pantalla.
8. Emisión de listas de precios así como también su modificación por incremento porcentual global o por línea de producto.
9. Listado de ventas por provincias y por artículo.

La capacidad de los archivos está definida por la fórmula siguiente:

129 x clientes + 54 x artículos + 152 x facturas mensuales.

El valor de esta fórmula no debe exceder 200.000.

Un ejemplo podría ser 1000 clientes, 300 artículos y 300 facturas mensuales, para este caso tenemos:

1000 cli. x 129 + 300 art. x 54 + 300 fact. x 152; esta cuenta da 190.800.

Esta fórmula contempla una cantidad de pagos similar a la cantidad de facturas.

## Gestión de compras

El sistema permite:

1. Ingreso de los datos de los proveedores.
2. Ingreso de los comprobantes emitidos por los acreedores así como también las propias Ordenes de pago.

Al ingresar el comprobante el sistema requerirá la imputación contable del mismo. Automáticamente preparará la información para el Libro de I.V.A. compras, el sistema contempla la corrección al I.V.A. compras por las Ordenes de Pago con bonificaciones.

3. Emitirá un listado que hace las veces simultáneamente de Libro de IVA, Subdiario de Acreedores e imputación contable, obteniéndose como totales al fin del listado la minuta correspondiente al asiento de compras del mes.
4. Consultar por pantalla la cuenta corriente de los proveedores interpretándola como saldo anterior más movimientos igual saldo actual.

La capacidad de archivos necesaria está definida por la siguiente fórmula:

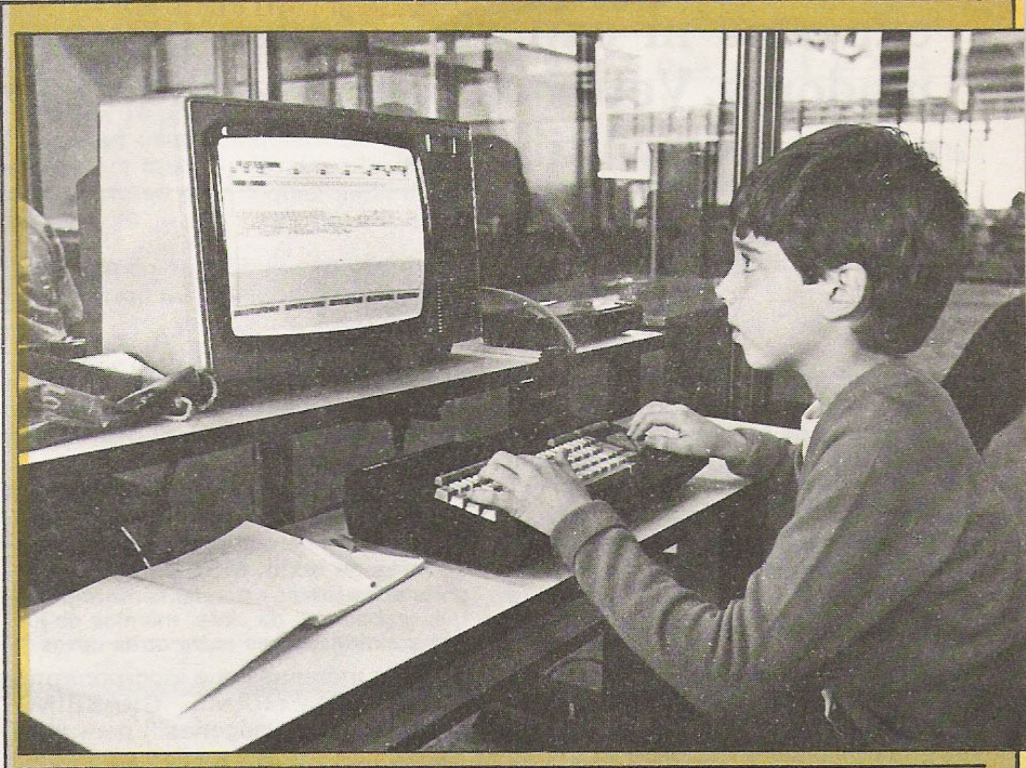
(Nota: Llamaremos a las cuentas de la contabilidad relacionadas con compras, simplemente cuentas, para abreviar la fórmula).

CAPACIDAD = Cuentas x 40 + proveedores x 127 + movimientos mensuales cuenta corriente x 72.

Esta capacidad no deberá exceder el número 220.000.

Como ejemplo supongamos el caso de tener 50 cuentas relacionadas con compras, 300 proveedores y unos 1.000 (mil) movimientos mensuales de cuenta corriente acreedores, la aplicación de la fórmula resultaría: 50 cuentas x 40 + 300 prov. x 127 + 1000 mov.cc.prov. x 72 = 112.000.

Como vemos por el resultado de este caso el sistema dispone de una gran capacidad de archivos para la mayoría de los casos posibles en las empresas.



# MSX - LOGO

• Talent es licenciataria exclusiva de Logo Computer System Inc., para el lenguaje MSX-LOGO, con adaptación de primitivas y redacción del Manual por los Ingenieros Hilario Fernández Long y Horacio Reggini.

El lenguaje Logo, creado especialmente en castellano para la Talent-MSX, ayuda a desarrollar la inteligencia, porque con MSX-LOGO los chicos aprenden haciendo. Esto es: aprenden razonando.

Con sus senderos exclusivos de entrada los lleva directamente hacia la idea central. Creando programas de lenguaje gráficos, colaboran en desarrollar su potencial intelectual y se expresan con libertad y espontaneidad.

El niño y el docente se convierten en maestros del computador. Porque una pregunta desata la próxima. Y el lenguaje LOGO está diseñado a partir de un principio: "sin umbral ni techo".

Se comienza a vivirlo sin experiencia previa en computación y su desarrollo posterior nunca es bloqueado por limitaciones en el lenguaje.

Por eso resulta el más adecuado para principiantes y para programadores expertos.

El computador, entonces, se incorpora hoy a la familia de los medios y recursos educativo-didácticos utilizados cotidianamente en el proceso de aprendizaje.

Según su creador, el Dr. Seymour Papert, "MSX y LOGO forman el matrimonio más ideal que podamos imaginar".

## Características:

- Gráficos de tortugas en 16 colores.
- Edita y almacena hasta 60 formas de tortuga.
- Opera con hasta 30 tortugas simultáneas.
- Detección de colisiones y mandos (Joystick).
- Animación, pintado y sombreado de gráficos.
- Uso optativo de ventana.
- Música y exploración de sonido.
- Matemática de alta precisión.
- Texto y gráficos en toda la pantalla.
- Editor de texto de pantalla completa.
- Procesamiento de palabras y listas.
- Listas de propiedades.
- Almacenamiento en discos o cassettes.

# METAMORFOSIS DEL BASIC

## Programar en Basic en cualquier computadora con sistema MSX no es para expertos ni experimentados programadores. Veremos aquí la facilidad de su manejo y utilidad.

Los computadores domésticos cuentan con gran canti dad de software pero a pesar de eso existen usuarios que no pueden disponer de ciertos utilitarios debido a que no se encuentran realizados para sus máquinas. Esto hace que el mercado de software como así también el de hardware (accesorios) se encuentre fragmentado.
Cuando uno se introduce en el campo de la computación y decide comprarse un ordenador, es difícil escogerlo, pues es necesario decidirse entre las distintas facilidades que nos brindan ciertas marcas.
Se debe tener en cuenta la capacidad de memoria, la potencia de su BASIC, capacidad gráfica y lo que tal vez sea más importante, su software (programas) y expansión.
No sólo importa saber la cantidad de programas que una marca nos brindará sino la calidad y utilidad de los mismos.
En otras palabras existen utilitarios muy potentes que son aplicables sólo para una porción muy pequeña de computadoras existentes en el mercado.
En 1983 dos firmas japonesas establecieron contacto con uno de los proveedores más importantes de software para micro-computadoras llamado MICROSOFT para que les escribiera una versión estandard de su BASIC. Poco tiempo antes otras doce compañías se acercaron a MICROSOFT con la misma propuesta. Esta unió todas las peticiones creando un intérprete BASIC optimizado y estandar que junto con reglas de fabricación y utilización de chips, forman el sistema MSX.
Por otra parte MICROSOFT anunció el lanzamiento del MSX-DOS, sistema operativo que amplía la capacidad de almacenar datos al permitir el uso de diskettes. Además es compatible con los sistemas operativos MS-DOS y XENIX.
MSX-DOS facilita el trabajo para aquellas personas que necesiten utilizar hojas de cálculo, planillas y procesadores de texto. MSX permite incorporar dispositivos estandar como impresoras, grabadores de cinta, mandos de juegos y sintetizadores entre otras cosas.
Existen máquinas que incorporaron la posibilidad "FRAME GRABBING" ("captador de imágenes") para almacenar una imagen de video y modificarla con gráficos nuevos que el usuario diseñe. Esta es una de las aplicaciones que se han realizado hace relativamente poco (en 1985) que nos asombra por ofrecer posibilidades inimaginables con los ordenadores de hace apenas un año.

### Un poco de HARDWARE.

Las computadoras MSX utilizan como CPU (Unidad Central de Proceso) el chip Z80A. La CPU es el procesador central encargado de que el ordenador realice las operaciones que el usuario le de por medio de un programa.
MSX utiliza el chip TMS9918A, que es el responsable de todo lo referido a gráficos que se verán en pantalla. Por último el sonido depende del chip AY-3-8910.
Estos tres chips forman parte del hardware del sistema MSX, no importa cuál sea la firma que fabrique el computador, siempre que haya adoptado este estandar.

### Manejo de gráficos y pantallas.

El chip de video TMS 9929A da la posibilidad de elegir 4 modos de pantalla; 2 de texto y 2 para gráficos. Cuando se enciende la computadora, el modo es 0. Este modo tiene 40 caracteres de ancho por 24 líneas. El modo 1 es de baja resolución para textos con 29 caracteres de ancho.



Pero la ventaja de este modo es que nos pérmite utilizar SPRITEs.
El modo 2 es de alta resolución para graficar. La pantalla se divide en 256 pixels de ancho por 192 de largo.
El modo 3 es el de baja resolución para gráficos que brinda una forma de gráficos por bloques multicolores.
El sistema MSX cuenta con 16 colores numerados del 0 al 15.
La Unidad Central de Proceso trabaja con 64k de RAM. Esta se encuentra mapeada en regiones bien definidas como el área de textos (donde se guardan los programas con sus números de líneas), área de variables y otras regiones. Pero el chip de video trabaja sobre otros 16k de RAM. Esto significa que los gráficos no se almacenan en las mismas áreas donde se guardan las variables definidas por el usuario o los programas.

## Dibujos animados en nuestro ordenador.

El Video Display Processor (VDP) permite el movimiento rápido de SPRITEs. Los SPRITEs son caracteres gráficos definidos por el usuario que se mueven a través de la pantalla sin interrumpir la ejecución del programa.

Hay 4 medidas de Sprites: 8 por 8 pixels, 8 por 8 pixels ampliados, 16 por 16 pixels, 16 por 16 pixels ampliados. Se pueden usar cualquiera de los 16 colores por cada Sprite, y se los pueden posicionar en cualquier lugar de la pantalla.

Es posible animar dibujos mediante la combinación de gráficos Sprites gracias a la rapidez con que se ejecutan las operaciones con estos gráficos.
El manejo de Sprites es sencillo y no hace falta utilizar POKEs.
Microsoft introdujo en MSX BASIC dos comandos simples; PUT SPRITE para ubicar un gráfico SPRITE en la pantalla y SPRITE$ para definirlos.
El programa de la figura 1 nos muestra una de las cosas que se pueden hacer con SPRITEs.

## Puntos, líneas y figuras.

La gran capacidad de graficar en la pantalla es posible mediante los comandos LINE, PRESENT, POINT, CIRCLE, PAINT, PSET y DRAW.
Cada instrucción gráfica tiene varias opciones.

**FIGURA 1**

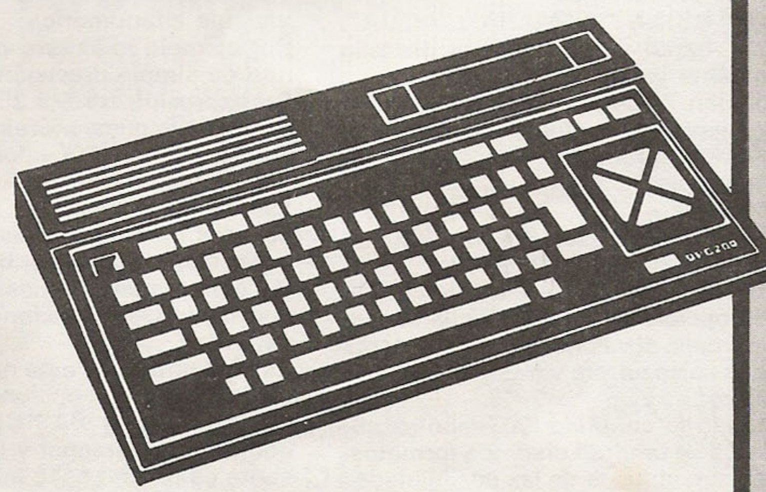
```
10 COLOR 15,4,7: SCREEN 1,3
30 FOR X=1 TO 32: READ W: X$=X$+CHR$(W):NEXT X: SPRITE$(1)=X$: X$=""
50 FOR F=0 TO 255
60 PUT SPRITE 1,(F,90),1
110 NEXT F
120 GOTO 50
130 DATA 1,3,3,1,7,11,19,11,7,3,195,70,44,24,0,0,128,192,192,128,192,196,232,208
,192,192,96,48,24,16,32,96
160 DATA 6,12,24,48,12,3,0,0,0,0,0,0,0,1,3,6,0,0,0,0,0,0,192,48,12,24,48,96,192,
128,0,0160
```


A la computadora personal
Talent MSX
nada le es imposible



Fernando L.
CORATELLA S.R.L.

COMPUTACION
HARDWARE - SOFTWARE



COSME BECCAR 249
(1642) SAN ISIDRO
Tel.: 743-0734/4962

Contado A 495 o en 3 cuotas de A 180

por ejemplo LINE puede dibujar rectángulos y cuadrados con la opción de resaltar el perímetro o el área con cualquier color. Este comando permite a su vez dibujar líneas rectas también en color. Mientras que con CIRCLE se realizan elipses, arcos, etc.
Pero si nosotros queremos diseñar gráficos más complicados tendríamos que usar la instrucción DRAW. Este comando utiliza el GRAPHICS MACRO LANGUAGE (GML) que con instrucciones simples como U, D, L, y R se puede dibujar líneas o mover hacia arriba, abajo, izquierda o derecha respectivamente. Con las órdenes E, F, G y H se dibuja o mueve en diagonal hacia arriba y a la derecha, abajo y a la izquierda. Todas estas opciones ván seguidas por un número que indica cuántos pixels deben dibujarse o moverse, contados desde la ubicación del último punto.
Si no se especifica el punto inicial de referencia, se tomará el ángulo superior izquierdo como punto de partida (0,0).
La opción Mx,y permite mover en forma relativa, esto quiere decir que depende del signo que precede a los valores "x" e "y". Por ejemplo DRAW "M-20,30" trazará una línea desde el

**FIGURA 3**

```
10 SOUND 11,8
20 SOUND 12,2
30 SOUND 6,20
40 SOUND 13,10
50 SOUND 0,6
60 SOUND 7,&B00000111
70 SOUND 8,16
```

último punto referenciado hasta un punto situado 20 pixels a la izquierda y 30 hacia arriba.
Por otro lado el comando Bn mueve el punto pero no dibuja ningún punto y Nn mueve y dibuja pero retorna a la posición original. Por ejemplo DRAW "BH20NE30" mueve el punto sin dibujar, 20 pixels en diagonal hacia arriba y a la izquierda, luego traza una línea de 30 pixels en diagonal hacia arriba y a la derecha, retornando 30 pixeles en diagonal hacia abajo y a la izquierda para volver a colocarse en la posición donde se encontraba antes de moverse los 30 pixels.
A n gira la figura formada con las instrucciones anteriores en intervalos de 90 grados. El 0 gira 0 grados, 1 gira 90 grados, 2: 180 grados y 3: 270 en sentido contrario a las agujas del reloj. Ejemplo: DRAW "A2U60" gira 270 grados el ángulo del dibujo, este ejemplo tiene el mismo resultado que DRAW "D60".
La opción Cn determina el color del gráfico; DRAW "C15U30" hará una línea de color blanco y de 30 pixels hacia arriba del último punto.
Y por último, Sn determina el factor de la escala, pudiendo tomar valores entre 1 y 255.

Propongo copiar el programa de la figura 2 y hacerlo correr.
Este es un ejemplo de cómo utilizar las opciones que analizamos recién.

## Efectos sonoros.

Si miramos ahora las posibilidades del sonido, son tan amplias como las de gráficos. La instrucción PLAY permite tocar música y con SOUND se pueden realizar efecto sonoros.
El generador de sonido permitirá usar hasta tres canales con la opción de utilizar especiales ondas envolventes.
El programa de la figura 3 simula el sonido de una locomotora.

## Una o más instrucciones en una tecla.

Las teclas de funciones se pueden redefinir.
Supongamos que queremos que se limpie la pantalla y que liste el programa en memoria, todo con sólo presionar una tecla.
Esto se consigue a través del comando KEY y con las respectivas órdenes.
De esta forma nuestro ejemplo habría que escribirse así: KEY 1, "CLS"+CHR$(13)+ "LIST"+CHR$(13). Observemos que el uno indica que F1 va a ser redefinida y después de cada instrucción debe estar la función de CHR$(13) que equivaldría al ENTER.

## Formatear salidas.

El comando PRINT USING imprime en la pantalla con el formato especificado en esta instrucción.
Para cadenas de caracteres, el signo de admiración "!" imprime sólo el primer caracter de cada cadena de la lista de expresiones.
PRINT USING "!";"MAPA"; "AGUA"; "LUZ" harán aparecer en la pantalla solamente las letras MAL.
La opción "& permite colocar variables de cadenas en cualquier parte de una constante caracter.
Por ejemplo
x$="lunes"
PRINT USING "Hoy & llueve"; x$
imprimirá: Hoy lunes llueve
Para imprimir números se puede utilizar la opción "##.###" que sacará por pantalla el valor con dos caracteres para la parte entera y tres para la parte decimal.
Variando la cantidad de símbolos numerales se tendrán distintos formatos.
Estas son algunas de las posibilidades de imprimir en la pantalla con formato.

## El tiempo también corre para nuestro computador.

Se tiene acceso al reloj interno de la máquina por medio de la variable TIME.
Esta variable se incrementa cada 1/50 segundos los equipados con sistema PAL de video y cada 1/60 segundos los provistos con sistema NTSC
La variable TIME no se incrementa cuando se está ejecutando una operación con cinta, quedando su valor sin aumentar durante ese tiempo. Pero una vez terminada la operación, continuará el conteo sin perder su valor.

## Variables y constantes

Por último vamos a hablar de las constantes numéricas y de las variables.
Las constantes numéricas se dividen en enteras y reales. Las enteras se pueden expresar en cuatro sistemas: decimal (0123456789), binario (01), octal (01234567) y hexadecimal (0123456789ABCDEF).
Las constantes numéricas reales se dividen en dos grupos: de simple precisión, si tienen menos de seis dígitos significativos; y de doble precisión para aquellos valores con siete o más números significativos pero con menos de catorce.
Los valores de las constantes alfanuméricas deben ir encerradas entre comillas ("). Ejemplo x$="sol"

**FIGURA 2**

```
10 SCREEN 2
20 PSET(143,96),0
30 FOR X=1 TO 4
40 FOR F=0 TO 3
50 A$="S"+STR$(X)+"C"+STR$(F+5)
60 A$=A$+"A"+STR$(F)+"E30"
70 A$=A$+"F40"+"G30"+"H40"
80 DRAW A$
90 NEXT F
100 NEXT X
110 GOTO 20
```

Para diferenciar las variables numéricas deberá agregarse algunos de los siguientes símbolos: % variable numérica entera; # variable de doble precisión; ! variable de simple precisión y $ variable alfanumérica.
Por ejemplo x! es una variable numérica de simple precisión.
La expresión x%=5.253 equivale a decir x=5 pues x% es una variable entera e ignora los valores decimales por truncamiento.
Las variables x%, x!, x$, x# son consideradas como variables distintas por tener cada una un símbolo diferente.
De esta manera hemos visto algunas instrucciones y funciones del BASIC estandar
Es fácil notar que este nuevo intérprete nos da más opciones y es más potente que el BASIC de cualquier computado personal y lo más interesante es que no hace falta instrucciones complicadas.
Leyendo el manual de la máquina y teniendo en cuenta lo que acabamos de comentar, estamos listos para realizar programas, siempre que le agreguemos a nuestros conocimientos un poco de creatividad.

# Talent MSX

## DISTRIBUIDORES OFICIALES

**COMPUPRANDO S.C.A.**
Av. de Mayo 965
(1085) Capital
Te.: 37-5159

**COMPUSHOP S.A.**
Córdoba 1464
(1055) Capital
Te.: 41-8730 - 42-9568
49-2165

**COMPUTRONIC S.R.L.**
Viamonte 2096
(1056) Capital
Te.: 40-4772/2279
46-6185

**CP 67 CLUB S.A.**
Florida 683 Loc. 18
(1005) Capital
Te.: 393-6303

**SERV. INFORMATICA S.A.**
Paraná 164
(1017) Capital
Te.: 35-1853/0832

**SIST. LOGICAL S.R.L.**
Esmeralda 561 3º B
(1007) Capital
Te.: 393-7669

(AREA EDUCATIVA
UNICAMENTE)

**DIST. CONCALES S.A.**
Tucumán 1458
(1050) Capital
Te.: 40-8664/0344

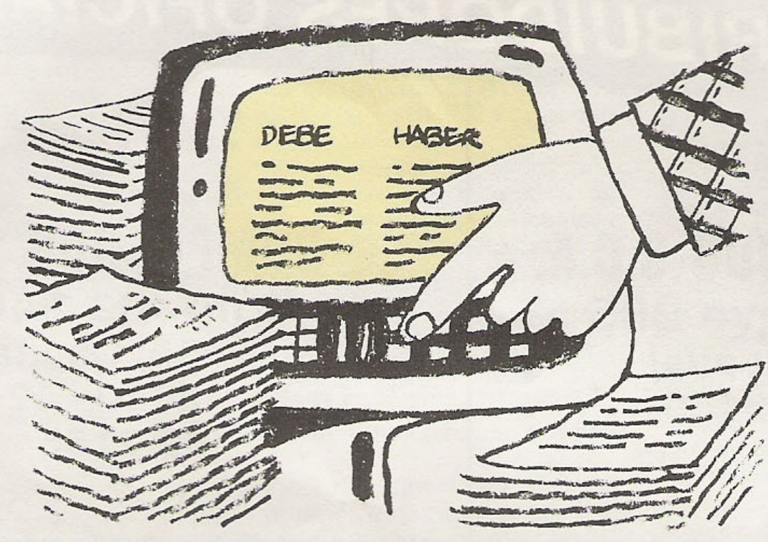
**MICROMATICA S.R.L.**
Av. Pueyrredón 1135
(1118) Capital
Te.: 821-5578




# ARCHIVO COMERCIAL

**Clase: comercial**

Es un programa que nos permite trabajar con un archivo para usarlos como agenda telefónica o en el rubro comercial para guardar datos de clientes, agregar firmas nuevas, borrar algunas o modificar información de otros.
Usa una estructura e instrucciones sencillas.
Modificando algunas líneas, podemos guardar los datos en cinta o en disco.
La línea encargada de esto es la 2030 para guardar y 2200 para recuperar.
En los OPEN de dichas líneas habrá que cambiar el nombre del dispositivo si se quiere guardar datos en disco. Basta solamente cambiar "cas" por "A"
El nombre "A" depende del que tenga nuestra disketera.
Inclusive el programa nos permite hacer un listado de los datos del archivo a través de la impresora.
Debemos tener conectado este dispositivo si elegimos la opción 4 del menú principal. En caso contrario la adorable maquinita se colgará haciéndonos perder los datos que se encuentren en memoria.
Si escogió la opción uno del menú principal, se le preguntará si los datos a entrar (o registros) pertenecen a un archivo nuevo. Si la respuesta es afirmativa, los registros almacenados en la memoria se borrarán para dar entrada a los del nuevo archivo.
En cambio si la respuesta es negativa, se agrandará a medida que agreguemos registros.
No podremos guardar más de 500 registros por archivo. Si el archivo necesita de más espacio este se consigue cambiando la dimensión de las matrices R$ y N$ por números más grandes a los que aparecen.
Los datos del archivo se pueden ver por medio de un listado en pantalla.
También se puede buscar información por el nombre del registro o por el contenido del mismo.
Supongamos estar trabajando con una agenda telefónica y queremos saber donde vive AGUSTIN ABRIL entonces entramos este nombre como nombre del registro y aparecerá toda la información de esta persona.
Pero si no sabemos de quién era el número telefónico, entonces lo entramos como información del registro y aparecerá el nombre del dueño como también toda su información.
Como pudimos observar las utilidades de este programa son muy amplias.
Entonces no perdamos más tiempo y pongámonos a tipear el programa que no es tan largo.




## Estructura del programa:

- **170-180**: Inicializa los arreglos o matrices para los nombres de los registros y para la información.
- **190-340:** prepara y muestra el menú principal
- **360-400:** toma y convalida la respuesta
- **410-590**: crea registros nuevos
- **610-920**: corrige registros
- **930-1030:** borra registros
- **1050-1080:** corrige el nombre
- **1090-1130:** corrige información
- **1150-1190:** agrega información
- **1210-1310:** lista por pantalla los registros con su información
- **1330-1430:** lista los registros por impresora
- **1450-1840:** busca registros por el nombre o por información
- **1860-1960:** ordena los registros
- **1970-2100:** carga desde cinta los datos
- **2120-2270:** guarda los datos en cinta
- **2290-2300:** subrutina de retraso para que podamos leer la pantalla antes de volver al menú principal.
- **2320-2360:** subrutina de tecla presionada

## Variables: IMPORTANTES

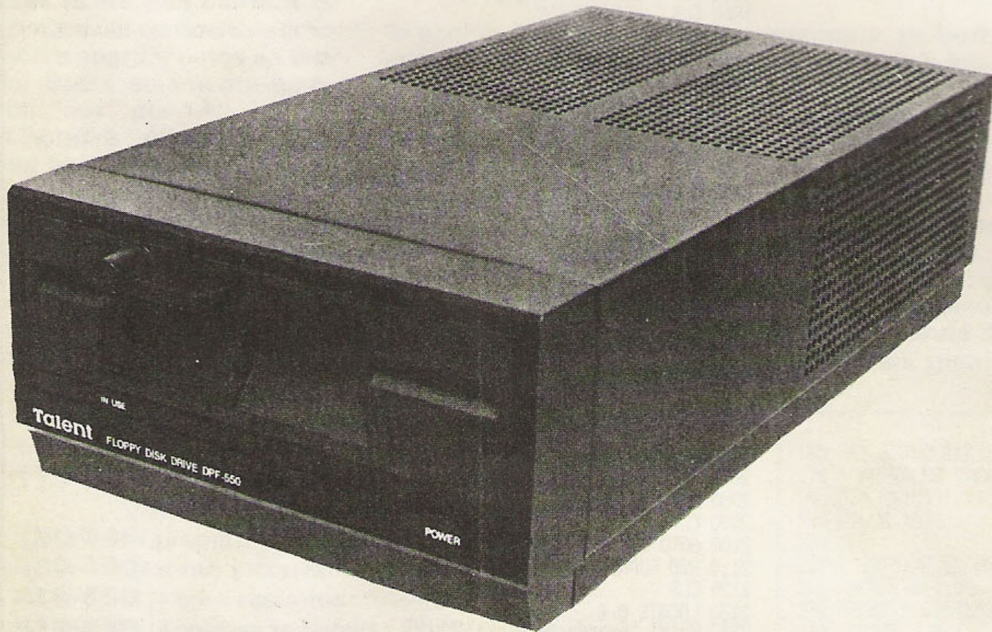
**R$:** matriz con los nombres de los registros
**N$:** matriz con los datos
**J**: cantidad de registros del archivo

```
160 KEY OFF:CLEAR 3000
170 ' ARREGLOS PARA GUARDAR NOMBRES
180 DIM R$(500),N$(500)
190 COLOR 15,1
200 CLS
210 ' IMPRIME EL MENU
220 LOCATE 6,1
230 PRINT " SISTEMA DE ARCHIVO HOGARE"+CHR$(&HA5)+"O"
240 LOCATE 0,4
250 PRINT"1. Ingresar nuevos registros":PRINT
260 PRINT"2. Corregir un archivo":PRINT
270 PRINT"3. Muestra los registros":PRINT
280 PRINT"4. Lista los registros en una           impresora":PRINT
290 PRINT"5. Busca un registro":PRINT
300 PRINT"6. Ordena los registros":PRINT
310 PRINT"7. Carga desde un cassette":PRINT
320 PRINT"8. Guarda en cinta":PRINT
330 LOCATE 0,23
340 PRINT"Seleccione numero de opcion..."
350 ' TOMA Y VALIDA LA RESPUESTA
360 A$=INKEY$
370 IF A$="" THEN 360
380 BEEP
390 IF A$<"1" OR A$>"8" THEN 360
400 ON VAL(A$) GOTO 420,610,1210,1330,1440,1860,1980,2120
410 ' REGISTRO NUEVO
420 CLS
430 LOCATE 0,6
440 INPUT "Es un nuevo archivo";Q$
450 IF LEFT$(Q$,1)="s" OR LEFT$(Q$,1)="S" THEN J=1 ELSE J=J+1
460 CLS
470 LOCATE 0,3
480 PRINT"Entre los datos - tipee 'FIN para    terminar"
490 PRINT:PRINT
500 LINE INPUT"Nombre del registro? ";N$(J)
510 IF N$(J)="FIN" OR N$(J)="fin" THEN 580
520 PRINT
```

```
530 LINE INPUT "Informacion? ";R$(J)
540 IF R$(J)="FIN" OR R$(J)="fin" THEN 580
550 PRINT:PRINT
560 J=J+1
570 GOTO 500
580 J=J-1
590 GOTO 200
600 ' CORRIGE REGISTOS
610 CLS
620 N=0
630 LOCATE 0,3
640 INPUT "Nombre del registro ";T$
650 PRINT :PRINT
660 M=1
670 IF T$=N$(M) THEN N=M:GOTO 740
680 IF M<501 THEN M=M+1
690 IF M<>J THEN 670
700 IF N<>0 THEN 720
710 PRINT"Registro con el nombre ";T$;"                  no existe"
720 GOSUB 2290
730 GOTO 200
740 PRINT N$(N):PRINT
750 PRINT R$(N):PRINT
760 INPUT "Es este registro? ";Q$
770 IF LEFT$(Q$,1)="s" OR LEFT$(Q$,1)="S" THEN 800
780 GOTO 680
790 ' MENU DE CORRECCIONES
800 CLS
810 LOCATE 0,4
820 PRINT"1. Borra un registro":PRINT
830 PRINT"2. Corrige el nombre de un registro":PRINT
840 PRINT"3. Corrige la informacion":PRINT
850 PRINT"4. Agrega informacion":PRINT
860 LOCATE 0,14
870 PRINT"Seleccione una opcion...":PRINT
880 A$=INKEY$
890 IF A$="" THEN 880
900 BEEP
910 IF A$<"1" OR A$>"4" THEN 880
920 ON VAL(A$) GOTO 940,1050,1100,1150
930 ' BORRA REGISTROS
940 FOR M=N TO J
950 N$(M)=N$(M+1)
960 R$(M)=R$(M+1)
970 NEXT M
980 N$(J)=""
990 R$(J)=""
1000 J=J-1
1010 PRINT"Registro borrado"
1020 GOSUB 2290
1030 GOTO 200
1040 ' CORIGE EL NOMBRE
1050 INPUT "Nuevo nombre: ";N$(N)
1060 PRINT:PRINT"Nombre cambiado"
1070 GOSUB 2290
1080 GOTO 200
1090 ' CORRIGE INFORMACION
1100 INPUT "Nueva informacion :";R$(N)
1110 PRINT:PRINT"Informacion cambiada"
1120 GOSUB 2290
1130 GOTO 200
1140 ' AGREGA INFORMACION
1150 INPUT "Nueva informacion";T$
1160 R$(N)=R$(N)+" - "+T$
1170 PRINT :PRINT "Nueva informacion agregada"
1180 GOSUB 2290
1190 GOTO 200
1200 REM MUESTRA LOS REGISTROS
1210 CLS
1220 D$=STRING$(36,219)
1230 FOR M=1 TO J
1240 PRINT: PRINT"Nombre del registro: ":N$(M)
1250 PRINT:PRINT R$(M)
1260 PRINT:PRINT D$
1270 FOR DE=1 TO 100: NEXT DE
1280 GOSUB 2320
1290 NEXT M
1300 GOSUB 2290
1310 GOTO 200
1320 REM LISTA LOS REGISTROS EN LA IMPRESORA
1330 CLS
1340 LOCATE 0,6
1350 LINE INPUT"Presione enter cuanado este lista la impresora"; NU$
1360 FOR M=1 TO J
1370 LPRINT
1380 LPRINT "Nombre del registro: ";N$(M)
1390 LPRINT
1400 LPRINT R$(M)
1410 LPRINT
1420 NEXT M
1430 GOTO 200
1440 REM BUSCA REGISTROS
1450 CLS
1460 LOCATE 0,6
1470 PRINT "1. Registro por el nombre":PRINT
1480 PRINT "2. Informacion string": PRINT
1490 LOCATE 0,12
1500 PRINT "Seleccione una opcion...":PRINT
1510 A$=INKEY$
1520 IF A$="" THEN 1510
1530 BEEP
1540 IF A$<"1" OR A$>"2" THEN 1510
1550 ON VAL(A$) GOTO 1560,1700
1560 INPUT "Nombre del registro"; T$
1570 PRINT
1580 F=0
1590 FOR M=1 TO J
1600 IF N$(M)<>T$ THEN 1640
1610 PRINT "Nombre del registro: "; N$(M)
1620 PRINT:PRINT R$(M)
1630 F=1
1640 NEXT M
1650 IF F=1 THEN 1670
1660 PRINT:PRINT "Registro no hallado"
1670 GOSUB 2290
1680 GOTO 200
1690 REM STRING
1700 INPUT "String a ser buscado ";T$
1710 F=0
1720 FOR M=1 TO J
1730 X=INSTR(N$(M),T$)
1740 IF X<>0 THEN 1770
1750 X=INSTR(R$(M),T$)
1760 IF X=0 THEN 1800
1770 PRINT:PRINT "Nombre del registro: ";N$(M)
1780 PRINT:PRINT R$(M):PRINT
1790 F=1
1800 NEXT M
1810 IF F=1 THEN 1830
1820 PRINT:PRINT" Registro no hallado"
1830 GOSUB 2290
1840 GOTO 200
1850 REM ORDENA LOS REGISTROS
1860 CLS
1870 LOCATE 0,6
1880 PRINT"Ordenando..."
1890 M=1
1900 IF N$(M)<N$(M+1) THEN 1940
1910 SWAP N$(M),N$(M+1)
1920 SWAP R$(M),R$(M+1)
1930 GOTO 1890
1940 M=M+1
1950 IF M<J THEN 1900
1960 GOTO 200
1970 REM CARGA DESDE CINTA
1980 CLS
1990 LOCATE 0,6
2000 PRINT"Carga desde cinta ":PRINT
2010 LINE INPUT "Presione return cuando el grabador   este listo";NU$
2020 PRINT:PRINT "Cargando..."
2030 OPEN"CAS:ARCHIVO"FOR INPUT AS 1
2040 INPUT #1,J
2050 FOR M=1 TO J
2060 INPUT #1,N$(M)
2070 INPUT #1,R$(M)
2080 NEXT M
2090 CLOSE
2100 GOTO 200
2110 REM GUARDA EN CINTA
2120 CLS
2130 LOCATE 0,6
2140 PRINT "Guarda en cinta":PRINT
2150 LINE INPUT "Presione return cundo este todo listopara grabar";NU$
2160 PRINT:PRINT" Grabando..."
2170 MOTOR ON
2180 GOSUB 2290
2190 MOTOR OFF
2200 OPEN "CAS:ARCHIVO" FOR OUTPUT AS 1
2210 PRINT#1,J
2220 FOR M=1 TO J
2230 PRINT #1,N$(M)
2240 PRINT #1,R$(M)
2250 NEXT M
2260 CLOSE
2270 GOTO 200
2280 REM SUBRUTINA DE RETRASO
2290 FOR DE=1 TO 2500:NEXT DE
2300 RETURN
2310 REM SUBRUTINA DE TECLA PRESIONADA
2320 Z$=INKEY$
2330 IF Z$="" THEN 2360
2340 Z$=INKEY$
2350 IF Z$="" THEN 2340
2360 RETURN
```

# LA PODEROSA DISKETERA DPF-550

**Uno de los avances más grandes de la Talent MSX lo constituye su capacidad de almacenamiento de datos. Y para ésto han desarrollado la disketera DPF-550, que es manejada enteramente por el sistema MSX-DOS, que hasta el momento es la mejor versión existente para microcomputadoras.**




Veremos aquí detallada y sencillamente algunas de las funciones y comandos que nos permite el MSX-DOS.

El DOS (Disck Operative System) es un sistema operativo creado por Microsoft al igual que el sistema MSX. Esto explica que esta versión sea superior a la de cualquier otra computadora que haga gala de su uso.

Este sistema se dedica exclusivamente al manejo de información.

Por trabajar exclusivamente con computadoras de 64 kbytes de RAM el MSX-DOS hace compatible el uso de datos de la IBM PC.

Y está compuesto por dos archivos, el COMMAND que es el procesador de comandos y por el MSXDOS que es el sistema operativo propiamente dicho. Ambos se encuentran en el disco que acompaña a la disketera.

Recordemos que este sistema nos permite usar de sus lujos tanto del MSX-DOS como desde el MSX-BASIC. Por esto comenzaremos viendo las instrucciones que se nos ofrecen para usar en MSX-BASIC.

**BLOAD:** funciona igual que en cassette y posee las mismas variantes.

**BSAVE:** ídem anterior.

**CLOSE:** Su uso es muy simple. Su función es cerrar archivos abiertos con la función Open.

Por ejemplo: Close #1 cierra el archivo número 1. Si se usa la instrucción CLOSE sin número de archivo, todos los que hayan sido abiertos son cerrados.

También se pueden cerrar algunos de los archivos con CLOSE #1,#2...

**COPY:** Su función es copiar todo tipo de archivos. Permite copiar uno, algunos o todos los archivos de un disco en otro, con su nombre original o uno nuevo.

Ejemplos:

1) COPY "A:*.*" TO "B:"; copia en el disco que se encuentra en el segundo drive (llamado B) todos los archivos que se encuentren en el disco del drive A.

2) COPY "A:Lalo" TO"B:"; carga el archivo Lalo en el disco del drive B con el mismo nombre. De no tener otro drive asumirá que se trata de copiar en el mismo drive, por lo tanto cuando nos pida que introduzcamos el disco en el drive B, simplemente tendremos que cambiar el disco que tenía el archivo a copiar por el nuevo disco con el mismo drive.

Para detener esta operación debemos pulsar Control-Stop cuando nos pida cambio de disco.

**DSKO:** Su función es copiar un bloque de memoria en un sector determinado del disco.

Recordemos que un disco está formado por Tracks que son los círculos concéntricos del disco (como los surcos de un disco musical) y cada Track está dividido en Sectores de 512 bytes.

Este comando cargará entonces desde un bloque de memoria a cuyo comienzo deben apuntar el par de direcciones &HF351 y &HF352, el contenido de éste en el sector especificado.

Ejemplo: POKE &HF351,64: POKE &HF352,156 hace que las direcciones que mencionábamos antes apunten a la posición 40000.

Luego de cargar los 512 bytes que siguen a la posición 40000 (40000 a la 40511) con la información que queremos guardar, podemos entrar DSKO (número de drive), (número de sector).

El número de drive corresponde a 0 para que asuma el drive que está conectado, 1 para el drive A y 2 para el drive B.

El mínimo número de sector es 0. Hay que tener cierto cuidado con el sector elegido, pues podemos borrar información esencial para el drive. Por ejemplo si cargamos información nuestra en el sector 5 habremos borrado todos los nombres de los programas o archivos que poseíamos en ese disco.

**FIELD:** Su función es la de reservar una cantidad determinada de bytes en un buffer destinado a la disquettera.

La mejor forma de entender su funcionamiento es con un ejemplo así que aquí va uno.

Supongamos que en un disco tenemos un programa Basic llamado CONSUELO, y queremos ver de qué forma lo ha guardado nuestro Drive.

Para ésto deberíamos hacer un programa como el de la figura 1. Veamos como funciona.

En primer lugar la línea 10 toma el programa CONSUELO como un archivo

aleatorio, o mejor dicho, abre un canal (#1) a través de la unidad de discos para leer un archivo (CONSUELO) en forma aleatoria, sin importarle su condición de programa Basic.
Recordemos que un archivo de esta naturaleza permite, a diferencia de uno secuencial, leer registros llamándolos con un número o subíndice en cualquier orden. Para comprender mejor esto último podemos suponer que un archivo abierto en forma aleatoria, es tratado como una matriz donde cada elemento (registro o renglón) tiene una longitud determinada. Dicha longitud la determina el comando FIELD.
Siguiendo con el análisis del programa, la línea 20 reserva un espacio de 10 bytes al cual le da el nombre B$.
De este modo cuando queramos leer uno de los registros que componen a CONSUELO, usaremos la instrucción GET # (número de archivo con que se abrió), (número de registro a leer) como figura en la línea 40.
Es importante mencionar que el espacio que reserva en el buffer la instrucción FIELD, puede variar entre 1 y 256 bytes, y que el número que definamos no puede superar la longitud del archivo que hallamos abierto.
La línea 30 inicializa un bucle, que nos mostrará cada uno de los registros que componen el archivo.
La instrucción de la línea 40 permite que de la DPF pase al bufer, una "tira" de 10 bytes (por estar definido así por la instrucción FIELD) del archivo presente en el canal 1, para luego hacer referencia a ella con el nombre B$.
La línea 50 imprime en pantalla el contenido de la "variable" B$ (tener en cuenta que no es una variable como la conocemos, sino el nombre de un campo en el buffer destinado al MSX DOS).
Por último la línea 60 cierra el bucle y el canal 1 que corresponde al archivo CONSUELO.
Con respecto al "Número máximo de registros" que figura en la línea 30, podremos conseguirlo teniendo en cuenta la longitud total del archivo (en bytes) sobre (/) la longitud del espacio reservado en el buffer.
Otras posibilidades de la función FIELD (campo) es la de definir estos espacios en forma de matriz o vector. Buenos ejemplos de esto se hallan en la página 75 del manual del drive.

**FILES:** Nos permite listar todos o algunos de los archivos existentes en el directorio de un disco.
Ejemplos:
FILES sin parámetro listará todos los archivos presentes en el directorio del disco actual.
FILES "*.DAT" listará los nombres de los archivos que tengan la extensión DAT.
**FILES**"*"listará los nombres que no tengan extensión del directorio.

**LFILES:** Su funcionamiento es el mismo que el del anterior, pero provoca la salida por impresora.

**FORMAT:** Nos permite inicializar el disco. Hay que tener en cuenta que ésto provoca el borrado de toda información si es que éste la contenía.
Por medio de esta instrucción el MSX-DOS prepara al disco para soportar 360 kbytes de información organizados en 80 Tracks con 9 sectores de 512 bytes por track.

**GET:** Nos permite leer un registro de un archivo abierto como de uso aleatorio. Como parámetros deberemos darle el número con que se abrió dicho archivo y el número de registro que queremos cargar en el buffer. Un buen ejemplo de su uso podemos verlo en la explicación del comando FIELD.

FIGURA 1

```
10 OPEN "CONSUELO" AS #1
20 FIELD #1,10 AS B$
30 FOR F=1 TO NO MAX REG.
40 GET #1,F
50 PRINT B$
60 NEXT F:CLOSE#1
```

**INPUT #:** Permite el ingreso de datos de un archivo abierto para trabajar en forma secuencial, a una variable.
Se puede leer tanto datos numéricos como alfanuméricos. En el primero de los casos los espacios, caracteres de control de carro, y el caracter de line feed son ignorados al realizar la lectura. El primer caracter encontrado luego de éstos, será tomado como valor numérico. El fin del dato numérico debe ser alguno de los caracteres que decíamos que eran ignorados, antes del dato numérico.
Lo mismo vale para datos alfanuméricos, con las siguientes salvedades:
1) Si el caracter encontrado es una comilla, tomará todos los caracteres que sigan a éste como un string hasta la siguiente comilla.
2) Si el string no comienza con una comilla, se asumirá el final del mismo cuando se encuentren algunos de los famosos caracteres (line feed, etc.), una coma o al exceder los 255 caracteres.
Si el archivo se termina durante la carga de datos numéricos o de tipo caracter, ésta concluye.
Ejemplo:

```
10 OPEN "EJEMPLO.DAT" FOR IN
PUT AS #1
20 INPUT#1,A$
30 PRINT A$
40 CLOSE #1
```

**KILL:** Este es el último de los comandos que veremos por ahora. Se encarga de asesinar al programa o archivo que encerremos entre comillas luego de esta instrucción.
Ejemplos:
KILL "TAMARA.*" borra del disco todos los archivos llamados TAMARA que tengan extensión.
KILL "*.ASP" borra del disco en curso, todos los archivos cuya extensión sea ASP.
Seguramente habremos notado que el * se utiliza como comodín, para los casos en que no recordamos un nombre o extensión, o cuando se requiere aplicar un comando a un grupo determinado de archivos.
El segundo comodín disponible es el signo de interrogación, y se coloca en el lugar de la letra que no recordamos.
En la próxima entrega continuaremos viajando por el magnífico mundo del MSX-DOS y la DPF-550.
Seguramente después de este avance, si no tenemos la disquetera es porque estábamos leyendo otra revista, ¿no?

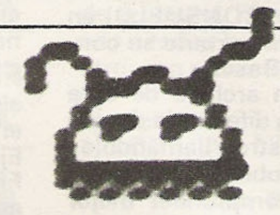







# EL DEVORADOR

## Clase: juego



### Estructura del programa:

| | |
|---|---|
| **20-120** | definición de sprites |
| **130-240** | pantalla de presentación |
| **250-310** | dibuja el laberinto |
| **320-370** | inicializa variables y posiciona los sprites |
| **390-860** | mueve los bichos y acepta los movimientos con los joysticks. |
| **880-1030** | subrutina para comer |
| **1050-1110** | pantalla final y resultados |

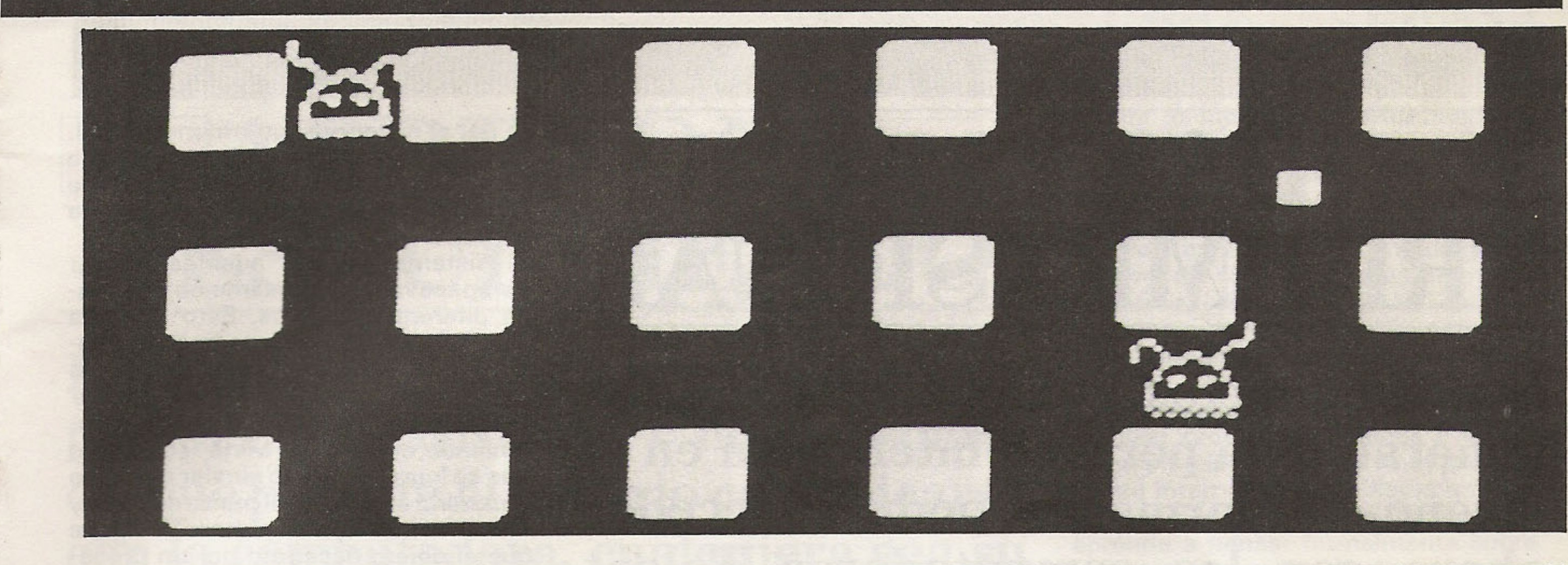
### Variables importantes

**B$:** tiene el valor alfanumérico de los sprites
Este valor es borrado en la línea 80
**S (1):** guarda el movimiento del jugador uno
**S (2):** guarda el movimiento del jugador dos
**P1:** puntaje del jugador uno
**P2:** puntaje del jugador dos.

**Es un juego sencillo pero entretenido y bastante corto para copiar. Permite la participación de dos personas con joystick. Cada jugador es un pequeño devorador que tiene que tratar de llegar a la comida antes que su contrincante lo logre.**
**Pero para complicar un poco las cosas nos encontramos en un laberinto que cambia cada vez que alguno de los bichitos consiguió devorarse el banquete. Hay que tratar de moverse bastante antes que el laberinto nos rodee con cuatro paredes.**

**Para comer no alcanza con llegar primeros a la posición donde se ubica la comida. Como somos bichos de pocos dientes necesitamos quedarnos unos segundos masticando en el lugar.**
**Si este paso lo obviamos, perderemos la oportunidad de ser nosotros los devoradores y nuestro contrincante podrá sacarnos el manjar que nos esperaba. Por eso hay que tener paciencia y masticar lo más rápido que se pueda. Y no olvidemos movernos continuamente para no quedarnos encerrados.**

```
10 SCREEN 2,2:COLOR 6,15,15:CLS
20 REM Definicion de sprites
30 FOR A=0 TO 1
40 FORB=1 TO 32
50 READ A$:B$=B$+CHR$(VAL(A$))
60 NEXT B
70 SPRITE$(A)=B$
80 A$="":B$=""
90 NEXT A
100 ' Data de sprites
110 DATA 128,128,128,96,16,9,14,8,16,22,34,32,32,63,42,21,0,0,0,6,9,145,113,16,8
,104,36,4,4,252,168,84
120 DATA 0,0,0,96,144,137,142,8,16,22,36,32,32,63,42,21,1,1,1,6,8,144,112,16,8,1
04,68,4,4,252,168,84
130 OPEN"GRP:"AS#1
140 COLOR13:PRESET(60,20):PRINT#1,"EL DEVORADOR"
170 PRESET(60,97):COLOR10:PRINT#1,"JUGADOR 1 (JOY.PORT .1)"
180 PRESET(60,127):COLOR 6:PRINT#1,"JUGADOR 2 (JOY.PORT 2)"
190 PRESET(46,180):COLOR1:PRINT#1,"PRESIONE ALGUNA TECLA"
200 PUT SPRITE1,(30,120),4:PUT SPRITE0,(30,90),12
210 FOR T=1 TO 25:A$=INKEY$:A=RND(1):IF A$<>"" THEN 260 ELSE NEXT T
220 PUT SPRITE 0,(30,120),4:PUT SPRITE 1,(30,90),12
230 FOR T=1 TO 25:A$=INKEY$:A=RND(1):IF A$<>""THEN 260 ELSE NEXT T
240 GOTO 200
250 'Dibuja el laberinto
260 PUT SPRITE 0,(19,0),15:PUT SPRITE 1,(19,0),15:CLS
270 FOR A=2 TO 256 STEP 32
280 FOR B=2 TO 192 STEP 32
290 LINE(A+1,B+1)-(A+15,B+15),6,BF
300 PRESET(A+1,B+1):PRESET(A+15,B+1):PRESET(A+1,B+15):PRESET(A+15,B+15)
310 NEXT B,A
320 'Variables
330 X=19:Y=129:Q=131:W=17:P1=0:P2=0:TIME=0
340 PUT SPRITE 0,(X,Y),12
350 PUT SPRITE1,(Q,W),4
360 GX=INT(RND(1)*7):GY=INT(RND(1)*5)
370 LINE(GX*32+24,GY*32+24)-(GX*32+28,GY*32+28),11,BF
380 ' Loop principal
390 S(1)=STICK(1)
400 IF TIME>4950 THEN 1050
410 ON S(1) GOTO 470,420,520,420,570,420,620,420
420 S(2)=STICK(2)
430 ON S(2) GOTO 670,390,720,390,770,390,820,390
440 IF POINT(Q+8,W+8)=11 THEN P1=P1+1000:GOTO 880
450 IF POINT(X+8,Y+8)=11 THEN P2=P2+1000:GOTO 880
460 GOTO 390
470 IF POINT(X+8,Y-8)=6 THEN 420 ELSE IF Y<10 THEN 420
480 FOR A=Y TO Y-16 STEP -1
490 PUT SPRITE 0,(X,A),12
```



```
500 NEXT A
510 Y=Y-16:GOTO 420
520 IF POINT(X+24,Y+8)=6 THEN 420 ELSE IF X>222 THEN 420
530 FOR A=X TO X+16
540 PUT SPRITE0,(A,Y),12
550 NEXT A
560 X=X+16:GOTO 420
570 IF POINT(X+8,Y+24)=6 THEN 420 ELSE IF Y>153 THEN 420
580 FOR A=Y TO Y+16
590 PUT SPRITE 0,(X,A),12
600 NEXT A
610 Y=Y+16:GOTO 420
620 IF POINT(X-8,Y+8)=6 THEN 420 ELSE IF X<19 THEN 420
630 FOR A=X TO X-16 STEP -1
640 PUT SPRITE0,(A,Y),12
650 NEXT A
660 X=X-16:GOTO 420
670 IF POINT(Q+8,W-8)=6 THEN 390 ELSE IF W<10 THEN 390
680 FOR A=W TO W-16 STEP -1
690 PUT SPRITE1,(Q,A),4
700 NEXT A
710 W=W-16:GOTO 390
720 IF POINT(Q+24,W+8)=6 THEN 390 ELSE IF Q>222 THEN 390
730 FOR A=Q TO Q+16
740 PUT SPRITE1,(A,W),4
750 NEXT A
760 Q=Q+16:GOTO 390
770 IF POINT(Q+8,W+24)=6 THEN 390 ELSE IF W>153 THEN 390
780 FOR A=W TO W+16
790 PUT SPRITE 1,(Q,A),4
800 NEXT A
810 W=W+16:GOTO 390
820 IF POINT(Q-8,W+8)=6 THEN 390 ELSE IF Q<19 THEN 390
830 FOR A=Q TO Q-16 STEP -1
840 PUT SPRITE 1,(A,W),4
850 NEXT A
860 Q=Q-16:GOTO 390
870 ' Para tomar la comida
880 FOR R=1 TO 80 STEP 10
900 NEXT R
910 LINE(GX*32+24,GY*32+24)-(GX*32+28,GY*32+28),15,BF
920 GX=INT(RND(1)*7):GY=INT(IND(1)*5)
930 IF POINT(GX*32+24,GY*32+24)=6 THEN920
940 LINE(GX*32+24,GY*32+24)-(GX*32+28,GY*32+28),11,BF
950 FOR A=1 TO 6
960 LINE(TX(A)*32+20,TY(A)*32+20)-(TX(A)*32+32,TY(A)*32+32),15,BF
970 NEXT A
980 FOR A=1 TO 6
990 TX(A)=INT(RND(1)*8):TY(A)=INT(RND(1)*6)
1000 IF POINT(TX(A)*32+24,TY(A)*32+24)<>15 THEN 990
1010 LINE(TX(A)*32+20,TY(A)*32+20)-(TX(A)*32+32,TY(A)*32+32),6,BF
1020 NEXT A
1030 GOTO 390
1040 ' Fin del juego
1050 CLS
1060 COLOR11:PRESET(60,55):PRINT#1,"SU PUNTAJE";P1:COLOR 6:PRESET(60,85):PRINT#1
,"SU PUNTAJE";P2
1070 PRESET(72,180):COLOR 13:PRINT#1,"OTRA VEZ (S/N)?"
1080 PUT SPRITE 1,(30,50),4:PUT SPRITE 0,(30,80),12:FOR T=1 TO 10:A$=INKEY$:IF A
$<>"" THEN 1100 ELSE NEXT T
1090 PUT SPRITE 0,(30,50),4:PUT SPRITE1,(30,80),12:FOR T=1 TO 100:A$=INKEY$:IF A
$<>"" THEN 1100 ELSE NEXT T
1100 IF A$<>"S" AND A$<>"N" THEN 1080
1110 IF A$="S" THEN CLS:GOTO 140
1120 END
```

# SISTEMA TRIDIMENSIONAL

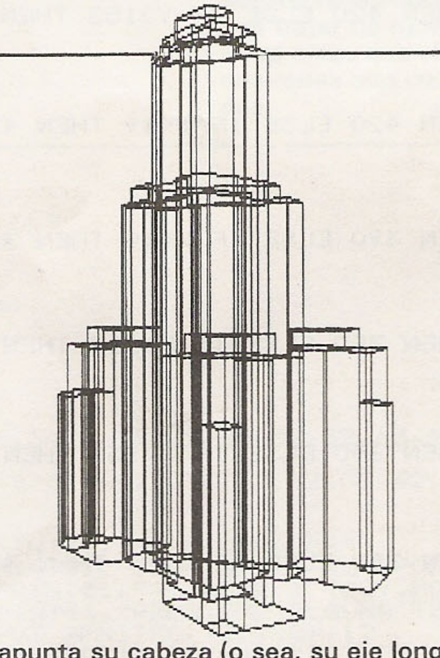
**Los arquitectos e ingenieros - y, en general, toda persona interesada en el diseño de formas - podrán acceder ahora con las computadoras Talent MSX a un novedoso sistema que posibilita la realización de dibujos tridimensionales de manera muy simple y natural.**

Este sistema, diseñado por el Ing. Horacio Reggini y explicado en detalle en su libro "Ideas y Formas. Explorando el espacio con Logo" de reciente aparición, se basa en la idea de que un objeto cualquiera puede ser descripto indicando, a partir de un punto, los movimientos que deberían hacerse para recorrerlo en su totalidad. Así cualquier persona, sin necesidad de contar con conocimientos matemáticos complejos y avanzados, puede realizar rápidamente cualquier diseño espacial.

La sencillez del sistema está intimamente ligado al lenguaje utilizado para la descripción de las formas: el Logo. Logo, de amplia difusión en el ámbito educativo y empleado en la actualidad por miles de niños de todo el país, representa una vertiente muy rica dentro del mundo de la computación.

Su implementación en equipos cada vez más poderosos y la aparición de sistemas como el que estamos describiendo le están abriendo las puertas de los mundos científicos y profesionales.

Es sabido que una parte del vocabulario Logo está dedicada a proporcionar medios simples para el tratamiento de problemas geométricos. A tal fin, emplea recursos distintos de los tradicionales, los que en su conjunto se denominan "geometría de la tortuga". Los dibujos se describen como si fueran realizados por una "tortuga"; esta, al moverse, objetiva nuestra idea acerca de cómo realizar una forma. Al desplazarse según el itinerario impuesto por un procedimiento, la tortuga deja un rastro o huella, realizando así la figura deseada. La ubicación espacial de la tortuga Logo se define por la posición que ocupa en un determinado instante y por la orientación hacia la cual



apunta su cabeza (o sea, su eje longitudinal).

En Logo, la descripción de una forma cualquiera se almacena en la computadora bajo un nombre; la escritura de ese nombre - a modo de orden o frase imperativa - hace que aparezca en la pantalla de la computadora el dibujo del objeto descripto. El "idioma de la tortuga", al asemejarse a un lenguaje humano natural, favorece ciertas analogías y modos de pensar.

Hasta ahora ha sido usual en Logo la realización de figuras planas. Este sistema permite a la tortuga escapar del plano de la pantalla y moverse en el espacio, creando así formas tridimensionales. Estas formas, generadas por la estela que deja la tortuga al moverse en el espacio, se proyectan en la superficie plana de la pantalla de modo similar a cómo los objetos espaciales aparecen usualmente en las pinturas o en las fotografías. La representación en el plano de la pantalla de los objetos que describimos en el espacio es efectuada automáticamente por el sistema Logo tridimensional, siguiendo las reglas de la perspectiva central clásica, desde un punto de vista fijo y situado frente al centro de la pantalla.

El sistema permite además dibujar perspectivas de un mismo objeto desde diferentes ángulos. Esto se logra desplazando a la tortuga hacia distintos lugares de la pantalla o haciéndola girar antes de generar la imagen. De esa manera, movemos el objeto manteniendo el punto de vista fijo, con lo que se logra un efecto similar al que se obtendría moviendo el punto de vista y manteniendo fijo el objeto. A fin de que el objeto descripto por un procedimiento aparezca en la pantalla, debe estar ubicado en alguna posición dentro del cono geométrico delimitado por el punto de vista y el marco de la pantalla, lo que en fotografía equivale a situar el objeto dentro del cono visual cuyos límites son el centro óptico de las lentes y los bordes de la película sensible. Esto se logra, ya sea apuntando la cámara fotográfica hasta ver el objeto en la forma y posición deseada, o bien manteniendo la cámara fija y moviendo el objeto, que es lo que se hace en este caso.

La posición y orientación iniciales de la tortuga en el espacio definen, en relación con el punto de vista, la imagen del objeto tridimensional que una orden crea. Así, por ejemplo, si previo a dar una orden acercamos la tortuga al punto de vista, la imagen se agrandará. Inversamente, será más pequeña si alejamos la tortuga del punto de vista.

La forma de proceder de este sistema desarrollado por el ingeniero Reggin reviste características que la acercar más a la modalidad operativa de ur artesano antes que a un trabajo planificado y lógico, ya que la interacción con la máquina es muy fluída y el plan de trabajo sufre constantes modificaciones durante su realización. Al igual que el artista, el artesano realiza frecuentes pausas durante la realización de su obra y, de acuerdo con lo obtenido, a veces la corrige o replantea su plan. Se podría decir que el material o la obra en elaboración guían y orientan su mano. Los instrumentos utilizados por el artesano no son complejos ni requieren conocimientos sofisticados para poder ser manejados. Necesitan sí de la habilidad y de la calidad estética del artesano. La computadora es un instrumento como lo es un cincel o un pincel y se emplea como vehículo para la expresión humana, ya sea de naturaleza tecnológica o artística, brindando la oportunidad de experimentar la emoción y la alegría del acto creativo. Adquiere eficacia solo en manos de personas con la imaginación, la maestría y el buen gusto necesarios para la concreción de cualquier obra.

El sistema descripto 30 - Logo está disponible para ser inmediatamente utilizado en el MSX-Logo de las computadoras Talent MSX.

# PROGRAMACION EN CASTELLANO

**El LPC fue creado hace poco más de dos años por un grupo de profesionales argentinos, los que basados en experiencias de otros países, entendían que existía un vacío no sólo dado por la barrera idiomática sino por las características de los lenguajes disponibles.**

Es conocido por todos que, si bien es el BASIC el lenguaje más difundido, ingresar a través de él a la programación puede traer consecuencias desagradables por ser un lenguaje que permite un desarrollo desorganizado, dado que no es estructurado. En contraposición el PASCAL o el ALGOL son lenguajes estructurados que llevan a una disciplina para la resolución de problemas, pero resultan muy complejos para un principiante e ininteligibles para un niño o adolescente. Si a esto le sumamos una barrera idiomática, nos encontramos ante la necesidad de crear un lenguaje en castellano, con la simpleza del BASIC, pero estructurado y con la potencia del PASCAL o el ALGOL (por ejemplo la recursividad) y además que sea divertido y ameno (por ejemplo con manejo de sonido, colores y dibujos), ésto es el LPC, un lenguaje que reúne todas estas características y que a diferencia de otros, no se cierra en sí mismo sino que permite la continuidad a través de los lenguajes estructurados (PASCAL, ALGOL, C, etc.) que son el presente y no pueden ser ignorados.

El LPC va dirigido al principiante, dispone de elementos como para que grandes y chicos puedan aprender jugando y jugar aprendiendo. Se pueden desarrollar dibujos y darles color y movimiento, también manejar sonidos con sus notas, todo a través de sus nombres (ROJO, AZUL, DO, RE, MI, etc.) sin necesidad de emplear o conocer códigos extraños.

Puede, a través del LPC, encararse la resolución de problemas sencillos con un método organizado y estructurado de resolverlos. Cuando nos referimos a la resolución estructurada de un problema, estamos indicando que debemos partir de lo general a lo particular. El problema es enfrentado en forma global, subdividiendo el todo en partes o pasos y a partir de allí efectuar el análisis particular de cada parte como un todo, hasta obtener elementos o estructuras simples indivisibles, las cuales en la síntesis (su rearmado en una estructura compleja) nos den la solución del problema.

En lo referente al área educativa, LPC se adapta tanto para los más pequeños en la primaria como para los estudiantes secundarios que se inician en el tema. Ya se han dictado seminarios para docentes primarios y secundarios en distintos colegios y para supervisores de educación media.

El intérprete LPC se encuentra hoy disponible en la más variada gama de microcomputadores hogareños y profesionales, más del 90% de los equipos disponibles en plaza pueden correr el LPC, destacándose las versiones para CP/M, MSDOS y MSX. Siendo la versión MSX una de las más potentes, podría decirse que en algunos aspectos supera inclusive a la versión MSDOS.

Lic. José Luis Ferreyro

## INSTRUCCIONES L P C

**SI-OTRO-FIN:** Chequea una condición y si esta es verdadera ejecuta un grupo de instrucciones, y en caso de ser falso ejecuta otro.
**ECO:** Ejecuta una cantidad N de veces un grupo de instrucciones.
**ECO HASTA:** Se ejecuta hasta que se cumpla una condición.
**USE:** Para utilizar variables o para definir operaciones matemáticas.
**COLOR:** Da color a una figura o al fondo de la pantalla.
**DIBUJE:** Se usa para dibujar una forma predefinida con la instrucción FORME.
**FORME:** Sirve para crear figuras
**SONIDO:** Emite notas musicales
**ESCRIBA:** Sirve para escribir sobre la pantalla constantes alfanuméricas o valores de variables.
**INGRESE DATO:** Se usa para ingresar datos desde teclado (variables numéricas)
**INGRESE TEXTO:** Ingresa desde teclado datos para variables alfanuméricas
**BORRE:** Se usa para borrar la pantalla
**PARE:** Detiene la ejecución del programa
**ENTERO:** Asigna el valor entero de una expresión a una variable
**ASIGNE:** Asigna y concatena variables alfanuméricas
**GRAFICOS:** Habilita la pantalla gráfica
**FIN-GRAFICOS:** Retorno a pantalla de texto
**EJECUTE:** Llamado a una subrutina
**RUTINA:** Rótulo de la subrutina
**VUELVA:** Retorna el control al programa principal

# APRENDER DESCUBRIENDO

## El Logo es un lenguaje fácil de comprender, en castellano y no matemático, que permite al usuario comunicarse con la computadora, cualquiera sea su nivel de conocimientos y su experiencia.

LOGO es un lenguaje de computación basado en una filosofía de la educación, desarrollado durante 12 años por Seymour Papert y el laboratorio de Inteligencia Artificial del Massachusetts Institute of Technology. Muchas de sus premisas están basadas en la teoría del conocimiento de Piaget, quien describe la evolución intelectual del niño a través de una serie de estadios del pensamiento.

Durante estos estadios, los actos cognitivos son el resultado de la forma de adaptación organizada del niño a su ambiente tal como él lo percibe. En cada estadio la comprensión del mundo por parte del niño se amplía, con la incorporación de nuevas experiencias, a las estructuras ya existentes y las modifica.

Para Piaget, el factor importante es que cada niño debe acumular una cierta cantidad de experiencias en cada nivel del pensamiento.

Las computadoras pueden ser portadoras de experiencias, pueden ayudar a establecer nuevas relaciones entre los campos, diferenciados artificialmente, de la ciencia, lo humanístico y del conocimiento de uno mismo. "Pero dice Seymour Papert, hay un mundo de diferencia entre lo que las computadoras pueden hacer y lo que la sociedad elegirá hacer con ellas".

La capacidad de resolver problemas, de ordenar secuencias lógicas, de tomar decisiones y pensar por sí mismo son patrones de comportamiento que se desarrollan explorando la modalidad LOGO.

Se puede trabajar por ensayo y error, razonando sintética o analíticamente, por anticipación de la idea, desarrollando planes totales o parciales. Estas alternativas apuntalan, amplían y fortalecen el proceso de aprendizaje.

Se pueden utilizar procedimientos puros, es decir, que pueden cerrarse y usarse como módulos. Cada parte es una "porción del pensamiento" que puede integrarse en otra mayor. Esto facilita la construcción, la asimilación y la transmisión del conocimiento.

LOGO posibilita la simulación de ambientes con los cuales se puede explorar, ricos en experiencias potenciales. Estimula el aprendizaje a través del descubrimiento, que es la forma natural de aprender y ordenar la realidad.

Permite al usuario, cualquiera sea su nivel de pensamiento y su experiencia, comunicarse con la computadora usando un lenguaje fácil de comprender, en castellano y no matemático.

Posibilita la extensión, ya que los procedimientos creados pasan a ser nuevas órdenes del lenguaje. Permite la interacción, dado que la respuesta puede ser inmediata y la corrección simple. Permite la recursión, todo procedimiento puede llamarse a sí mismo.

El trabajo con LOGO no tiene como finalidad aumentar el coeficiente intelectual o mejorar la performance de tests estandarizados. Por el contrario el objetivo es estimular al usuario a ampliar conceptos ya existentes en sus estructuras de pensamiento a través de la exploración y manipulación, y por medio de esta tarea lograr la concretización de áreas del conocimiento que previamente parecían ser muy abstractas, teóricas o inobservables.

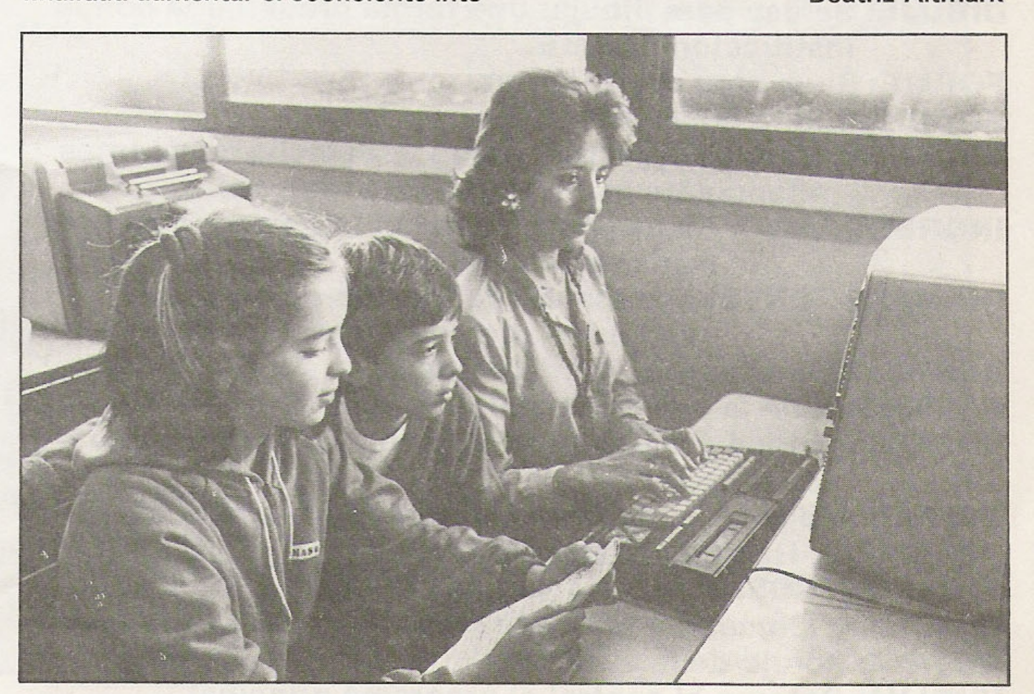
La "Geometría de la tortuga" permite resolver problemas gráficos a través de un principio heurístico muy poderoso: se internaliza el problema a resolver asumiendo el rol de Tortuga. Se aprende a pensar formalmente sobre la matemática, referida al propio cuerpo, usándola como un lenguaje más.

Para realizar un proyecto específico, el alumno tiene que comunicarse con la computadora, para que ésta interprete qué hacer y cómo hacerlo. Es el sujeto el que determina el nivel de complejidad que desea explorar en el área de resolución de problemas o en el área de la comunicación. Es el alumno el que controla la computadora, la que a su vez responde reforzando sus respuestas.

La computadora no pretende reemplazar al maestro o a los métodos tradicionales de enseñanza. Debe ser considerada como una herramienta para asistir, servir de apoyo y facilitar el aprendizaje. Es el maestro quien debe seleccionar el estilo de enseñanza que mejor se ajuste al estilo de aprendizaje de sus alumnos.

Como advierte Paul Torrance: "Si no confío en el alumno deberé intentar ahogarlo con información que yo elija para evitar que siga un camino equivocado. Pero si tengo confianza en la capacidad del individuo para desarrollar sus propias potencialidades, puedo darle la oportunidad de elegir su camino y su propia dirección en el aprendizaje" (Orientación del talento creativo, ed. Troquel, Bs. As. 1969).

Beatriz Altmark



# Talent MSX

## DISTRIBUIDORES OFICIALES

**F. CORATELLA S.R.L.**
Cosme Beccar 249
(1642) San Isidro
Te.: 743-0734

**HOT BIT COMP.**
Carlos Casares 997
(1712) Castelar Norte
Te.: 629-2247

**MANIAC COMP.**
Av. Rivadavia 13734
(1704) Ramos Mejía
Te.: 654-6844

**ARGESIS COMP. S.A.**
Meeks 269
(1832) Lomas de Zamora
Te.: 243-1742

**SERV. INFORMATICA S.A.**
Av. del Libertador 882
(1638) V. López
Te.: 35-1853

**MICROSTAR S.A.**
Eduardo Costa 892
(1640) Acassuso
Te.: 45-0964

**CERO-UNO INFORMATICA S.A.**
Calle 48 Nº 529
(1900) La Plata
Te.: (021) 24-9905/9906
24-9907

**MINICOMP**
Tinogasta 4044
(1417) Capital
Te.: 50-2535/7713




# AGILIZANDO GRAFICOS

**Para simular por ejemplo que un cuadro se mueve por la pantalla, es necesario un paquete de instrucciones, para que dibuje el cuadro, lo posicione en la pantalla, se mueva un lugar y que finalmente borre el primer cuadro. Pero esto se simplifica si definimos al protagonista del paseo como un sprite.**

En muchos programas de juegos aparecen hombres, naves espaciales y toda clase de bichos moviéndose por la pantalla. Estas figuras nos entretienen porque pueden desaparecer y volver a aparecer o detenerse, y en el caso de las naves espaciales hasta explotan.

Estos gráficos se llaman SPRITE y se definen como gráficos móviles monocromos.

Un sprite no puede ser tapado por ninguna letra o adibujo que no sea otro sprite. Por eso si en el centro de la pantalla hay una hormiga definida como sprite, la única forma para dejar de verla será poniendo encima otro sprite.

La pantalla que usa la computadora está formada por capas o planos paralelos de display. El plano más cercano al observador lleva el número cero. Todas las imágenes que estén en el plano cero, parecerán pasar por adelante de las imágenes de los demás planos. El plano más lejos al observador es el plano principal, donde comunmente estan los textos y gráficos. (ver figura 1).

El plano próximo al observador tiene prioridad más alta sobre los restantes. Esto significa que un sprite del segundo plano es tapado por uno que esté definido en el plano cero; pero el del segundo plano puede tapar a otro del plano tres hasta uno del plano treinta y uno. Entonces el sprite del plano trigésimo p rimero no podrá tapar ningún diseño de este tipo.

Hay treinta y dos planos de display (numerados del cero al treinta y uno) y por cada plano puede haber solo un sprite; esto explica por que normalmente no es posible tener más de treinta y dos sprites en la pantalla al mismo tiempo.

Como podemos ver en la figura uno, después de los planos para sprites, esta el plano multicolor.

Aquí es donde se efectúan el manejo de gráficos y textos. Entonces las imágenes producidas con instrucciones como DRAW, LINE, PRINT, tienen menos prioridad que cualquier sprite, produciendo la sensación de que estas figuras moviles pasan por encima de un texto que se encuentra en la pantalla.

El último plano o "plano de fondo" es el que habitualmente vemos como borde de la pantalla.

Podemos usar estas figuras animadas en cualquier modo de pantalla menos en el modo cero.

## SPRITE

Los tamaños que pueden tomar son cuatro y cada uno correponde a un tipo de sprite. (Ver figura 2).

El tamaño más pequeño es entonces el de 8 x 8 pixels (tipo cero) y su forma sería como el de cualquier caracter en el modo 2 de pantalla; el más grande es el de 16 x 16 pixels ampliados (tipo tres).

El color del sprite puede ser uno de los 16 disponibles y se declara cuando posicionamos el dibujo en la pantalla.

El cuadro de la figura 3 representa 8 bytes (filas) de 8 pixels (casilleros) cada uno.

En el primer byte hay tres pixel sin pintar, tres pintados y dos sin.

Los pixels que carecen de color son transparentes. Por eso los pixels pintados de un sprite del plano tres podrán verse a travéz de los pixels sin

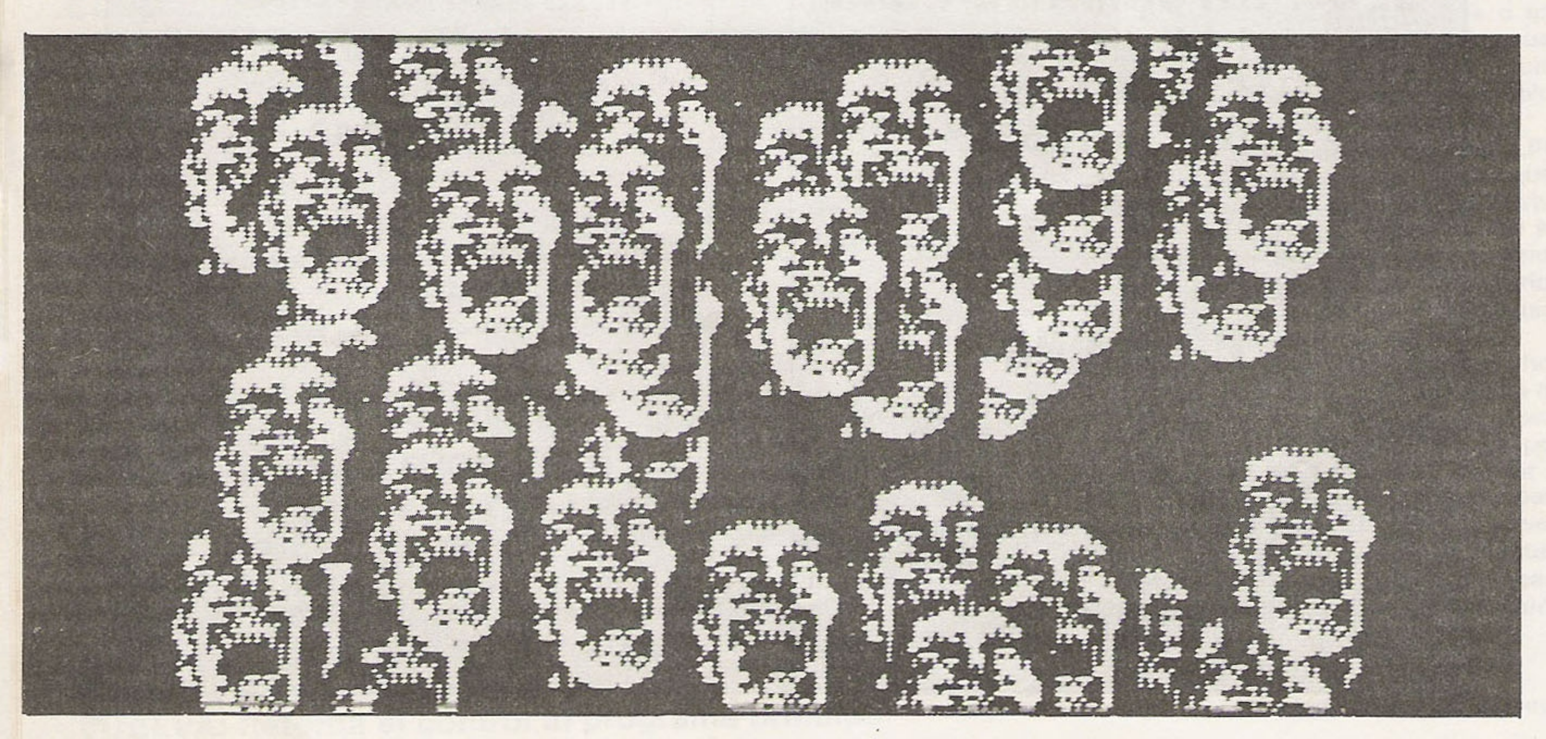




FIGURA 3

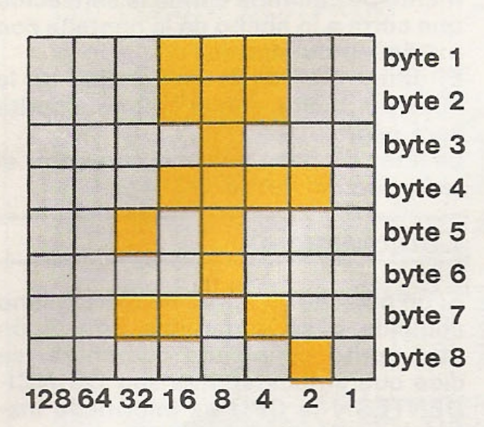



FIGURA 1




FIGURA 2

| TIPO | TAMAÑO |
|---|---|
| 0 | 8 x 8 pixels |
| 1 | 8 x 8 pixels ampliado |
| 2 | 16 x 16 pixels |
| 3 | 16 x 16 pixels ampliado |

pintar o transparentes de un sprite del plano dos por ejemplo, o de cualquier otro de prioridad superior como del plano uno o cero.

Todo casillero pintado lo marcamos con un uno y los casilleros transparentes con cero. De esta manera traduzcamos el dibujo de la figura 3 al sistema binario.

En la Figura 4 cada pixel corresponde a una potencia de dos (por estar en esa base) que va aumentado de derecha a izquierda. El primer byte se escribirá $2^0x0+2^1x0+2^2x1+2^3x1+2^4x1x$ $2^5x0+2^60+2^70$ y representa a sus ocho pixels escritos en forma decimal.

Si resolvemos la expresión entonces el primer byte tiene valor 28 en decimal y en binario sería 00011100 (el astuto lector se habrá dado cuenta que en binario (los valores) se comienzan a escribir de izquierda a derecha, al revés que en decimal.

El segundo byte tiene el mismo valor. El tercero es 00001000 en binario y 2 8 en decimal.

Asi el cuarto byte vale 30, el quinto 40, el sexto otra vez 8, el séptimo 52 y el octavo 2 en decimal.

Veamos los resultados del pasaje de sistemas en la figura cinco.

Una vez calculados los valores de cada fila, se los transfiere a la variable SPRITES.

Esta variable usa los siguientes parámetros: SPRITE$ (n) variable alfanumérica, donde n es un entero entre o y 255 (para los de tipo 0 y 1), o entre 0 y 63 para los de tipo 2 y 3.

La asignación de nuestra figura a un sprite, por ejemplo con el número 1 que corresponde al número del sprite, sería así:

50 x $=chr$ (28)+chr$ (28)+chr$ (8)+chr$ (30)+chr$ (40)+chr$ (8)+chr$ (52)+chr$ (2)

60 SPRITE$ (1)=x $

Así queda definido un sprite de 8 x 8 pixels mientras que uno de 16 x 16 se define como si fueran cuatro sprites de 8 x 8 y siguiendo el orden que se




## Ahorro de tiempo

En muchos de nuestros programas usamos bucles FOR...NEXT. Esta instruc-

```
10 CLS:TIME=0
20 FOR F%=1
   TO 2000
30 NEXT F%
40 PRINT TIME
```

```
10 CLS:TIME=0
20 FOR F=1
   TO 2000
30 NEXT F
40 PRINT TIME
```

ción va acompañada por una variable que si se mueve en un intervalo considerablemente grande pero con pequeños incrementos, el tiempo que tarda en realizar este bucle es el suficiente como para hacer desatar nuestras células nerviosas y terminar por cortar el programa. El ordenador toma la variable como si fuera de doble precisión ocupando el triple de espacio en la memoria que una variable entera y quitando velocidad a la ejecución del programa. Para evitar este inconveniente, se puede agregar el símbolo "%" (declara variables enteras) después del nombre de la variable.

Para notar bien la diferencia, comparemos los programas de la figura uno y dos.

Los dos programas realizan las mismas operaciones. En la línea 10 inicializamos el reloj interno en cero; las líneas 20 y 30 son bucles hasta 2000. En la última instrucción se imprime el tiempo que tardó en ejecutarse todo el programa.

Los dos resultados de la variable TIME no coinciden a pesar de la similitud de ambos programas.

Para ahorrar tiempo es aconsejable usar variables enteras siempre que el programa lo permita, definiéndolas al principio del programa.

## Lupa

Para lograr los caracteres en gran tamaño sin tener que redefinirlos, cambiemos el modo de pantalla con SCREEN 3, y hagamos imprimir algún mensaje como lo hace el programa de la figura seis. Pero el modo de pantalla

```
10 OPEN"GRP:" AS#1
20 SCREEN 3
25 PSET(0,50):PRINT#1,
   "* LOAD *"
30 PSET(70,120)
40 PRINT#1,"MSX"
50 GOTO 50
```

tres, no nos permite imprimir textos directamente entonces habrá que engañar al ordenador declarando un archivo en la pantalla con el número 1. Esto se define en la línea 10 de nuestro programa.

Las instrucciones PRINT van acompañadas por el símbolo que aclara que se imprimirá en un archivo, pero como éste se abrió sobre la pantalla, de ahí que se nos permita inscribir mensajes en el modo tres.

## Interrupciones

Si nos encontramos con algún programa largo, para ahorrar tiempo cortemos las interrupciones para leer el teclado y en consecuencia la actualización de la variable TIME.

Estas funciones si bien son realizadas con rapidez, tambien devoran nuestro tiempo. Con VDP (1)=208 se desactivan las lecturas del teclado. Aquí, el sistema no recibe ninguna instrucción desde el teclado, por lo tanto, las teclas STOP o CONTROL/STOP no provocarán ningún efecto.

Para volver a activar las lecturas del teclado, hace falta introducir VDP (1)=204.

muestra en la figura 6.
Entonces la definición sería:
60 SPRITE$ (1)=cadena del primer cuadrante+ cadena del segundo+ cadena del tercero + cadena del cuarto cuadrante.
El comando "SCREEN, modo, tipo", nos da a elegir el modo de pantalla y el tipo de sprites (ver tipos de sprite de la figura dos) que usaremos
No se pueden usar dos o más tipos de sprite al mismo tiempo. La intrucción SCREEN inicializa todas las variables sprite$.
¿Cómo se ubica un sprite en pantalla?
Para posicionar un sprite en cualquier lugar de la pantalla se usa el comando PUT SPRITE número de plano, (x,y), color, número de patrón
El número de plano como vimos puede ser un valor entre cero y treinta y uno. La coordenada x esta comprendida entre -32 y 255 y la coordenada y entre -32 y 191 para los tres modos que soportan sprites.
Si el valor (x, y) es omitido, se posicionará el sprite en el último lugar donde accedió un comando gráfico. El punto de referencia de las coordenadas es el ángulo superior izquierdo donde "x" e "y" valen cero.
Para algunos valores de estas coordenadas se obtienen resultados inesperados que nos pueden ser muy útiles en los programas. Por ejemplo, si a un sprite le damos como valor de la coordenada."y"=208, todos los sprites de los planos display de prioridad inferior desapareceran de la pantalla hasta que "y" tome otro valor.
En cambio, para "y"=209 hará desaparecer el mismo sprite que lleva este valor para su segunda coordenada.
Haciendo variar los valores de los coordenadas de un sprite, dara la sensación que éste se desplaza suavemente por la pantalla.
Analizemos el programa de la figura 7
Figura 7

```
10 SCREEN 2,0
20 FOR X=1 TO 8: READ A: B$=B$+
CHR$ (A)
30 NEXT
40 SPRITE$ (1)=B$
50 FOR F=30 TO 190
60 PUT SPRITE 1, (F,100), 1
70 NEXT
80 GOTO 50
90 DATA 28, 28, 8, 30, 40, 8, 52, 2
```

La línea 10 elige el modo de alta resolución para gráficos y el tipo 1 para sprites (de 8 x 8).
Si hubiera elegido SCREEN 2,1 imprimirá el sprite ampliado.
En la línea 20 lee ocho valores del data de la línea 90 y los guarda en la variable alfanumérica B$.
La línea 40 permitirá, al ir incrementandose el FOR, que la posición del sprite especificada en la línea 60 aumente de columna dando la sensación que corre a lo ancho de la pantalla con movimientos suaves.
El último "1" de la instrucción de la línea 60, indica que el color del sprite será negro.
La línea 80 hace volver a comenzar el recorrido del sprite.

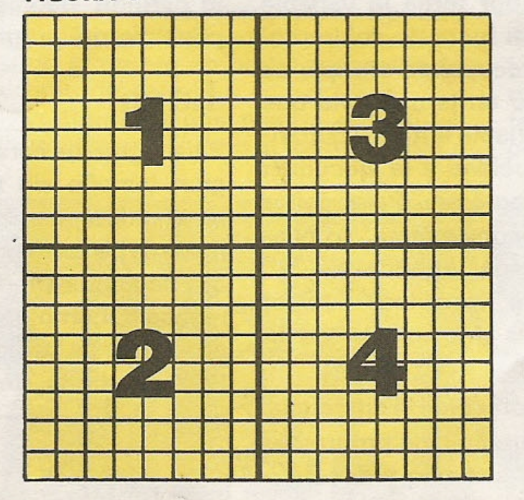
FIGURA 4

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

FIGURA 5

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | 16 | 8 | 4 | | | 28 |
| | | | 16 | 8 | 4 | | | 28 |
| | | | | 8 | | | | 8 |
| | | | 16 | 8 | 4 | 2 | | 30 |
| | | 32 | | 8 | | | | 40 |
| | | | | 8 | | | | 8 |
| | | 32 | 16 | | 4 | | | 52 |
| | | | | | | 2 | | 2 |

FIGURA 6




## Coincidencias

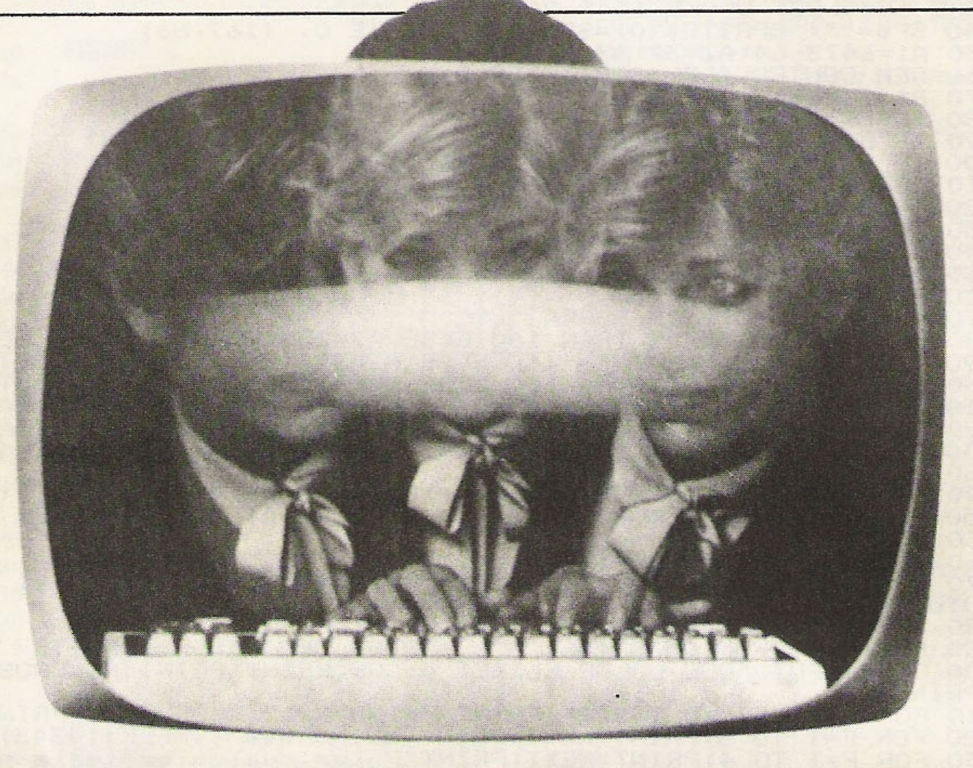
Si un pixel de un sprite de cierto plano coincide o se superpone con algún otro sprite de planos diferentes, se dice que ambos sprites son COINCIDENTES y la CPU es informada instantáneamente por el VDP.
Cuando esto ocurra, podemos ordenar al sistema ejecutar algun tratamiento útil para nuestro programa, como imprimir mensajes aclaratorios, borrar o hacer explotar al sprite que coincida.
Esto es posible siempre que esté activada la instrucción "ON SPRITE GOSUB n" (n es un número de línea donde slatará el programa si detecta una coincidencia).
Cada vez que esté esta instrucción y se produzcan coincidencias entre sprites, el sistema ejecutará la rutina especificada en el número n y regresará.
Pero si aun coinciden los sprite, se vuelve a entrar en la rutina otra vez casi inmediatamente, sin haberse realizado ningún otro proceso.
Para evitar la llamada repetida a la rutina es necesario desactivar el ON SPRITE con SPRITE OFF una vez que se consiguió llegar a la linea n, variar la posición de los sprites y lugar volver a activar la detección de coincidencias otra vez con ON SPRITE GOSUB.
Ahora sí podemos definir y hacer mover los sprites en la pantalla.
Todos los manejos de este tipo de gráficos se hacen con comandos sencillos como vimos.
Probemos graficar simples sprites de 8 x 8, movámoslos por la pantalla y veamos que diferencia hay entre SCREEN 2,2 y SCREEN 2,1.
La mejor manera de familiarizarnos con el sistema es probando y haciendo algunos cambios a los parametros.
El programa de fábrica de sprites nos permitirá diseñar gráficos.
Este programa nos dará los valores necesarios para poder definir nuestra creación en la variable SPRITE$.

```
10 REM ************************
20 REM *                      *
30 REM * FABRICA DE SPRITE *
40 REM *                      *
50 REM ************************
60 KEY OFF: COLOR 15,1,1:SCREEN 1,2:PRINT "  FABRICA DE SPRITES"
70 LOCATE 0,7: INPUT"      Que clase de sprite va adefinir: 8x8 (presione la
      tecla '8')                                              16x16 (presion
e'16')";A$
80 IF A$="16" THEN 460 ELSE IF A$<>"8" THEN 70
```




# DACTILOGRAFIA MSX

Clase: educativo



No hace falta ir a una academia para aprender a escribir a máquina.
Si alguna vez tuvimos la intención de escribir en el computador, sin necesidad de mirar el teclado y con dos dedos, aconsejamos tipear este programa que nos enseñará la ubicación de las letras y símbolos raros que existen en el mismo.
En el menú nos encontramos con la opción F1 que precisamente hará aparecer el menú.
La función F2 hará aparecer un cuadro que simulará la ubicación de las teclas que se nos vayan presentando. Mientras, en el margen inferior la letra pedida va avanzando y hay que tratar que no llegue al extremo izquierdo.
La frecuencia con que aparecen las letras es seleccionada con F5. Con esta función se aumenta o disminuye la rapidez.
Si presionamos
F4, cambia la opción entre mayúsculas o todo tipo de caracteres y símbolos.
En cambio con F3 la computadora nos exigirá que escribamos frases que deben ser primero introducidas por nosotros en el data de la línea 1150 en adelante.
Después de ajustar la elección con F3, F4, F5 habrá que presionar F2 para que nos ejercite tirándonos letras
Si no llegamos a contestar todas las tentativas correctamente y nuestro tiempo terminó, la máquina nos dará unos escasos segundos para que descansemos y nos tomará lección nuevamente.
Como opción, se podría guardar en una variable las veces que acertamos y en otra las que nos equivocamos. Esto nos permitiría saber nuestro nivel y ver si mejoramos.
Se podría agregar para esto la siguiente línea.
575 IF K$=INKEY$ THEN BIEN=BIEN+1 ELSE MAL=MAL+1:LOCATE 2,5:PRINT" BIEN"; BIEN, "MAL"; MAL

## Estructura del programa:

**240-320** : presenta el menú
**330** : inicializa variables (aquí hay que agregar BIEN=O y MAL=O si adherimos la línea 575)
**380-440** : presenta pantalla para jugar
**450-510** : elige una letra y la imprime en el tablero
**510-540** : demora según la velocidad especificada con F5
**550-560** : avanza la letra en el renglón inferior
**570-600** : convalida nuestra respuesta
**620-680** : subrutina para elegir algún símbolo raro como "%"
**740-750** : imprime la velocidad que habíamos elegido
**790-810** : imprime la opción según F4
**910-1190**: elige frases o palabras de los datas

## Variables IMPORTANTES:

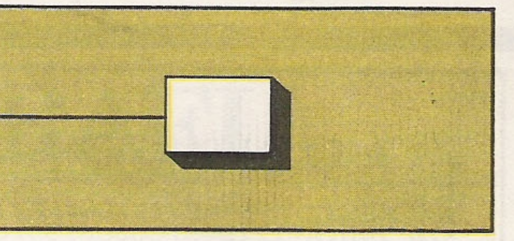
**K$:** letra que nosotros tecleamos
**N:** valor ASCII de la letra que está en cuestión
**SLOT:** da la velocidad

```
60 CLEAR 600:GOTO180
160 ON LISTGOS GOSUB 370,480,590,960
170 ON GOSUBS GOSUB 180,350,610,690,760,820,900
180 '
190 ON KEY GOSUB 180,350,900,760,690
200 KEY(1)OFF:KEY(2)ON:KEY(3)ON
210 KEY(4)ON:KEY(5)ON
220 CLS :COLOR15,4,7:SCREEN 0:KEY OFF
230 '
240 LOCATE 5,5
250 PRINT"TIM'S"
260 LOCATE8,22:PRINT"GHJDFH"
270 LOCATE 0,10
280 PRINT"F1 MENU":PRINT
290 PRINT"F2 CARACTERES":PRINT
300 PRINT"F3 PALABRAS/FRASES":PRINT
310 PRINT"F4 MAYUSCULAS/TODO TIPO DE CARACTERES":PRINT
320 PRINT"F5 AUMENTA VELOCIDAD/DISMINUYE VELOC.":PRINT
```

```
90 REM PREPARA LA PANTALLA DE                PRESENTACION
100 CLS:LOCATE 0,2 :PRINT" Pulse la tecla 'ESC' para       finalizar."
110 LOCATE 0,4:PRINT"   *---------*":FOR F=1 TO 8:PRINT"   *          *": NEXT: PRI
NT "   *---------*"
120 LOCATE 0,18:PRINT"   Use el '1' para fijar un      punto y '0' para borrar"
130 VPOKE 6484,24: VPOKE 6485,23: VPOKE 6486,25: VPOKE 6516,22: VPOKE 6518,22: V
POKE 6549,23: VPOKE 6548,26: VPOKE 6550,27
140 SP$="": SPRITE$(0)=SP$:PUT SPRITE 0, (167,86)
150 A1=6473-64:A2=32:A3=4:A4=4
160 REM CONTROLA EL SPRITE DE 8X8
170 B$=INKEY$: VPOKE A1,255
180 IF B$="" THEN 170
190 IF B$=CHR$(27) THEN 390
200 IF B$="1" THEN VPOKE A1,A2:GOTO 270
210 IF B$="0" THEN VPOKE A1,A2:GOTO 290
220 IF B$=CHR$(30)THEN 330
230 IF B$=CHR$(29) THEN 310
240 IF B$=CHR$(31) THEN 350
250 IF B$=CHR$(28) THEN 340
260 GOTO 170
270 REM MUEVE-DIBUJA-BORRA
280 IF VPEEK(A1)=255 THEN 160 ELSE A2=255:MA(A3,A4)=1: GOTO 370
290 IF VPEEK(A1)=32 THEN 160 ELSE A2=32: MA(A3,A4)=0: GOTO 370
300 IF VPEEK(A1)=32 THEN 160 ELSE A2=32: MA(A3,A4)=0: GOTO 370
310 IF A4<2 THEN 160 ELSE A4=A4-1: VPOKE A1,A2:A2=VPEEK(A1-1):A1=A1-1:GOTO 160
320 IF A4<2 THEN 160 ELSE A4=A4-1: VPOKE A1,A2:A2=VPEEK(A1-1):A1=A1-1:GOTO 160
330 IF A3<2 THEN 160 ELSE A3=A3-1: VPOKE A1,A2: A2=VPEEK(A1-32): A1=A1-32: GOTO
160
340 IF A4>7 THEN 160 ELSE A4=A4+1 :VPOKE A1,A2: A2=VPEEK(A1+1):A1=A1+1: GOTO 160
350 IF A3>7 THEN 160 ELSE A3=A3+1: VPOKE A1,A2: A2=VPEEK(A1+32): A1=A1+32: GOTO
160
360 REM MUESTRA EL SPRITE
370 B2=14335+A3:B= VPEEK(B2):B1=2^(8-A4): IF MA(A3,A4)=1 THEN B=B+B1:VPOKE B2,B:
GOTO 160
380 B=B-B1:VPOKE B2,B:GOTO 160
390 REM MUESTRA LOS DATOS          PARA DEFINIR EL SPRITE
400 CLS:INPUT "QUE COLOR ELIGE: ";COL:PRINT:PRINT"NUMEROS PARA EL DATA: ": PRINT
:PRINT
410 FOR I=14336 TO 14342: PRINT USING"###";VPEEK(I);:PRINT" ";:NEXT
420 FOR F=1 TO 3:PRINT:NEXT:PRINT USING"###";VPEEK(14343)
430 FOR F=1 TO 4:PRINT:NEXT:PRINT"Pulsa cualquier tecla para   volver a empezar"
440 PUT SPRITE 0,(127,150),COL
450 IF INKEY$="" THEN 450 ELSE RUN
460 REM PANTALLA PARA 16X16
470 CLS:PRINT "Pulsa la tecla 'ESC' para    finalizar":PRINT"*----------------*"
:FOR F=1 TO 16: PRINT"*                *":NEXT
480 A1=6474:A2=32:A3=8:A4=8
490 PRINT "*----------------*
500 A$="":DIM MA(16,16):SPRITE$(0)=A$:PUT SPRITE 0,(216,86)
510 VPOKE 6490,24:VPOKE 6491,23:VPOKE 6492,23:VPOKE 6493,25:VPOKE 6522,22:
520 VPOKE 6525,22: VPOKE 6554,22: VPOKE 6557,22: VPOKE 6587,23: VPOKE 6588,23:VP
OKE 6586,26:VPOKE 6589,27
530 REM MANEJA AL SPRITE
540 A$=INKEY$: VPOKE A1,255
550 IF A$="" THEN 540
560 IF A$=CHR$(27) THEN 770
570 IF A$="1" THEN VPOKE A1,A2: GOTO 650
580 IF A$="0" THEN VPOKE A1,A2:GOTO 660
590 IF A$=CHR$(30) THEN 670
600 IF A$=CHR$(29) THEN 680
610 IF A$=CHR$(31) THEN 690
620 IF A$=CHR$(28) THEN 700
630 GOTO 540
640 REM MUEVE-BORRA-DIBUJA
650 IF VPEEK(A1)=255 THEN 540 ELSE A2=255:MA(A3,A4)=1:GOTO 720
660 IF VPEEK(A1)=32 THEN 540 ELSE A2=32: MA(A3,A4)=0: GOTO 720
670 IF A3<2 THEN 540 ELSE A3=A3-1: VPOKE A1,A2: A2=VPEEK(A1-32):A1=A1-32:GOTO 54
0
680 IF A4<2 THEN 540 ELSE A4=A4-1: VPOKE A1,A2: A2=VPEEK(A1-1):A1=A1-1: GOTO 540
690 IF A3>15 THEN 540 ELSE A3=A3+1: VPOKE A1,A2: A2=VPEEK (A1+32): A1=A1+32: GOT
O 540
700 IF A4>15 THEN 540 ELSE A4=A4+1: VPOKE A1,A2: A2= VPEEK(A1+1): A1=A1+1: GOTO
540
710 REM MUESTRA EL SPRITE
720 IF A4<9 THEN B1=0 ELSE B1=16
730 B2=14335+A3+B1: B=VPEEK(B2): IF A4<9 THEN B1=2^(8-A4) ELSE B1=2^(16-A4)
740 IF MA(A3,A4)=1 THEN B=B+B1: VPOKE B2,B: GOTO 540
750 B=B-B1:VPOKE B2,B:GOTO 540
760 REM IMPRIME LOS                RESULTADOS DE LA DEF.
770 CLS: INPUT "Que color elige: ";COL:PRINT:PRINT:PRINT"Numeros para los data:
":PRINT"NO SE OLVIDE QUE HAY QUE     LEER ESTOS DATOS DE IZQUIERDAA DERECHA":PRI
NT
780 FOR F=14335 TO 14356 STEP 7:PRINT:PRINT:FOR J=1 TO 7: PRINT USING"###"; VPEE
K(F+J);:PRINT" ";:NEXT:NEXT
790 FOR F=1 TO 2 :PRINT:NEXT: FOR F=14364 TO 14367: PRINT USING"###"; VPEEK(F);:
PRINT" ";: NEXT
800 FOR F=1 TO 3: PRINT: NEXT:PRINT"Presione alguna tecla para    volver a empeza
r":PUT SPRITE 0,(104,50),COL
810 IF INKEY$="" THEN 810 ELSE RUN
```

```
330 SLOT=50:INC=10:CAPS=1
340 T=TIME:R=RND(-TIME):GOTO 340
350 '
360 KEY(2)OFF:KEY(3)ON:KEY(1)ON:CLS
370 Z=0:A=12:D=8:GOSUB730:GOSUB780
380 LOCATE A,D:PRINT" 43211 11234"
390 LOCATE A-2,D+1:PRINT"1"SPC(15)"1"
400 LOCATE A-2,D+2:PRINT"2"SPC(15)"2"
410 LOCATE A-2,D+3:PRINT"3"SPC(15)"3"
420 LOCATE A-2,D+4:PRINT"4"SPC(15)"4"
430 LOCATE A+2,D-2:PRINT"IZQ. DERE."
440 LOCATE 0,20:PRINT"["TAB(36)"]"
450 IF CAPS=1 THEN N=ASC("A")+INT(RND(1)*(ASC("[")-ASC("A"))) .
460 IF CAPS=-1 THEN N=ASC("!")+INT(RND(1)*(ASC("{")-ASC("!")))
470 IF N>90 AND N<97 THEN 460
480 GOSUB 610:' TECLAS DE POSICION
490 P=N-32:PLAY"L32V15N=P;"
500 LOCATE 18+Z,20:PRINT CHR$(N)
510 IF Z<0 THEN LOCATE A+A1,D+D1:PRINT CHR$(N)
520 IF TIME <T+SLOT THEN 520
530 IF TIME>10000 THEN TIME=0
540 T=TIME
550 LOCATE A+A1,D+D1:PRINT SPC(1)
560 LOCATE 18+Z,20:PRINT SPC(1)
570 K$=INKEY$
580 IF K$=CHR$(N) THEN Z=Z+1 ELSE Z=Z-1
590 IF Z=-18 OR Z=+18 THEN GOSUB 820
600 IF K$=CHR$(N) THEN 450 ELSE 490
610 '
620 RESTORE 680:D1=0
630 IF N=34 THEN A1=12:D1=3:GOTO 670
640 READ ST$:D1=D1+1
650 A1=INSTR(ST$,CHR$(N)):IF A1=0 THEN 640
660 D1=INT(D1/2+.5):'
670 RETURN
680 DATA !@#$%^&*() +,1234567890-=,QWERT YUIOP;qwert yuiop,"ASDFG HJKL:",asdfg h
jkl;',ZXCVB NM,./,"zxcvb nm,./"
690 '
700 ACROSS=POS(0):DAWN=CRSLIN
710 IF SLOT>100 OR SLOT<10 THEN INC=INC*-1
720 SLOT=SLOT*(100+INC)/100
730 '
740 LOCATE 0,0:PRINT INT(1/(SLOT/3000)+.5)"Ca/Min"SPC(13)INT(1/(5*SLOT/3000)+.5)
"Palab/Min"
750 LOCATE ACROSS,DOWN:RETURN
760 '
770 CAPS=-CAPS:ACROSS=POS(0):DOWN=CRSLIN
780 '
790 LOCATE 0,22
800 IF CAPS>0 THEN PRINT SPC(7)"(Letra Mayusculas solamente)"SPC(4)ELSE PRINT SP
C(7)"(Todos los caract. y simbol.)"
810 LOCATE ACROSS,DOWN:RETURN
820 '
830 FOR C=93 TO 1 STEP-3:L=INT(C/2)
840 C2=C+3
850 IF Z>0 THEN PLAY "V15O4L=L;M55S5R=L;C"
860 IF Z<0 THEN PLAY "V15L=L;R=L;N=C;","V15L=L;R=L;N=C2;"
870 NEXT:Q=7000
880 FOR C=0 TO Q:NEXT
890 Z=0:RETURN
900 '
910 P$="":KEY(3)OFF:KEY(1)ON:KEY(2)ON
920 RESTORE1140:T=TIME:CYCLE=RND(1)*5
930 READ P$
940 IF P$="END" THEN 920
950 IF TIME<T+CYCLE THEN 930
960 L=LEN(P$):P1$="":CLS:GOSUB 730
970 LOCATE 0,2:PRINT P$
980 FOR C=1 TO L
990 PLAY "V08L32F"
1000 IF TIME<T+SLOT THEN 1000
1010 IF TIME>10000 THEN TIME=0
1020 T=TIME:K$=INKEY$
1030 IF K$="" AND C=1 THEN 990
1040 P1$=P1$+K$
1050 NEXT C
1060 K$=INKEY$:IF K$<>"" THEN 1060
1070 LOCATE 0,11:PRINT P1$
1080 LOCATE 10,20
1090 IF P1$=P$ THEN PRINT "BIEN HECHO!!":ELSE PRINT "PRUEBE OTRA VEZ"
1100 IF P1$=P$ THEN Z=1 ELSE Z=-1
1110 GOSUB 820:'
1120 PRINT"BARRA DE ESPACIO=REPITE U OTRA TECLA=CORRE OTRA FRASE"
1130 K1$=INKEY$:IF K1$="" THEN 1130
1140 IF K1$=" "THEN 960 ELSE 920
1150 DATA "ESTA ES UNA FRASE"
1160 DATA "palabras nuevas"
1170 DATA "Palabras, oracion s de mas de 250     caracteres estan bien."
1180 DATA "Agregue otro texto data que desee     antes de terminar ."
1190 DATA "END"
```

# CONCURSO DE PROGRAMAS



**auspiciado por TELEMATICA S.A. que proveerá los siguientes Premios:**

## P R I M E R   P R E M I O

**Un periférico** (a elección entre un monitor, una diskettera y una impresora). **Una Beca** para trabajar en el Departamento de Investigación y Desarrollo de Telemática S.A.

## S E G U N D O   P R E M I O

**Un periférico** (a elección entre un monitor, una diskettera y una impresora).

En caso de que el ganador no pueda utilizar la beca será ofrecida a quien obtenga el segundo premio, y si este tampoco pudiera aprovecharla se otorgará a alguno de los participantes del certamen que se hubiera destacado.

## J U E G O S :

**Temas:**

- **TRUCO.** Premiaremos al programador que logre la mejor versión de este tradicional juego de salón.
- **DE INTELIGENCIA.** Los juegos que nos inspiren podrán ser "El Estanciero" o temas originales que sigan la línea

## E D U C A T I V O S :

**Tema:**

- **LIBRE.**

## U T I L I T A R I O :

**Tema:**

- **INTELIGENCIA ARTIFICIAL.** Se trata de lograr un programa que realmente nos haga discutir sobre el límite entre la simulación y la inteligencia de la computadora.

Obviamente la única forma de lograr esto será siguiendo las reglas propuestas de la Inteligencia artificial. Por esto se considerarán para premiar esta categoría, además de las reglas detalladas más abajo, la capacidad de auto-aprendizaje del programa, el nivel de inferencia del mismo, la capacidad y modo de almacenamiento de su base de datos, y principalmente su analizador sintáctico dado que hasta el momento no se ha logrado uno que dé suficiente credibilidad de que estamos frente a un ser racional que entiende nuestro idioma y no frente a una máquina a la que debemos hablarle con verbos en infinitivo al mejor estilo Tarzán (sin menospreciar a este último).

## PROFESIONAL O GESTION:

**Tema:**

- **LIBRE.** Dentro de este tema podrán figurar programas de las más diversas aplicaciones.

**BASES:** No sólo será indispensable que el programa enviado funcione correctamente sino que además debe cumplir con ciertas reglas.

- *Programación estructurada en bloques fácilmente diferenciables.*
- *Fácil seguimiento del mismo y detalle de éste como parte de su documentación. (Diagrama de bloques con los números de línea que los identifiquen).*
- *Aclaración y clara explicación de los algoritmos utilizados, deben figurar como parte de la documentación.*
- *Las variables y/o direcciones de memoria utilizados también se deben incluír en esta documentación.*
- *Listado de nemónicos assembler y la localización en memoria si es que se utiliza este tipo de lenguaje.*
- *Calidad y originalidad de gráficos, sonidos y pantallas de menú.*
- *El programa debe ajustarse a alguno de los temas propuestos más arriba. Esto es ELIMINATORIO.*

Los trabajos deberán enviarse antes del 30 de noviembre próximo (cierre del certámen) a: Paraná 720, piso 5°, (1017) Capital Federal.

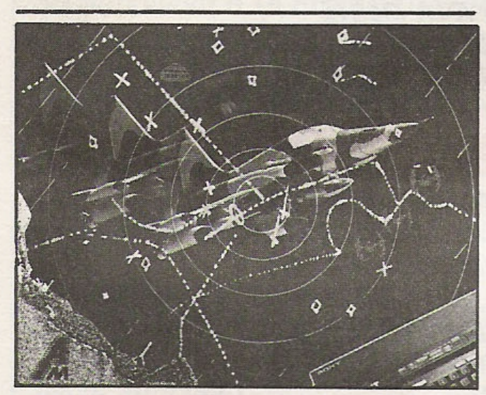



# Programación básica MSX
## Autor: Jonathan Pearce y Graham Bland
## Editorial: Paraninfo

Introducirnos en el nuevo mundo del MSX puede parecer algo Misterioso, Surreal y X.........; pero en este libro el manejo y utilidad del intérprete BASIC estandard está presentado con una explicación de fácil comprensión.
Nos introduce en el tema con datos muy elementales, explicados para todos los niveles, desde aquellos que recién empiezan hasta para los expertos programadores.
Las instrucciones se encuentran agrupadas según sean sus funciones. Así, el libro cuenta con sectores destinados a la explicación del empleo de gráficos, música, variables y constantes e interacción de programas.

Esta separación por grupos facilita el entendimiento de los comandos y nos ahorra el trabajo de buscar las instrucciones que podemos usar si nos proponemos realizar determinada tarea, como por ejemplo tocar la Quinta Sinfonía de Beethoven en nuestro computador.

# El libro gigante de los juegos para MSX
## Autor: A Lancey
## Editorial: Anaya Multimedia



Este libro cuenta con una variedad de juegos que estimulan nuestra imaginación para poder crear un juego de cualquier tipo. Cada juego está completamente explicado. Desde su objetivo hasta el algoritmo que se utiliza para la ejecución y sus gráficos.
Además, cada programa está acompañado por alguna sugerencia para modificarlo y obtener de esta forma una versión más particular.
En el libro están utilizadas prácticamente todas las instrucciones y a ellas podemos recurrir si no entendemos la utilización de algún comando.
La mayoría de los programas son cortos y en consecuencia fáciles de copiar. Pero para hacer más sencilla esta tarea, este trabajo contiene una subrutina que ayuda a rastrear errores. Todos los que alguna vez copiamos DATAs extensos, sabemos qué tedioso es buscar un número mal tipeado en estas sentencias.

CompuPrando

Av. de Mayo 965
1084 - Bs. As.
Tel.: 38-0295

HARDWARE
.EQUIPOS
.PERIFERICOS
.ACCESORIOS

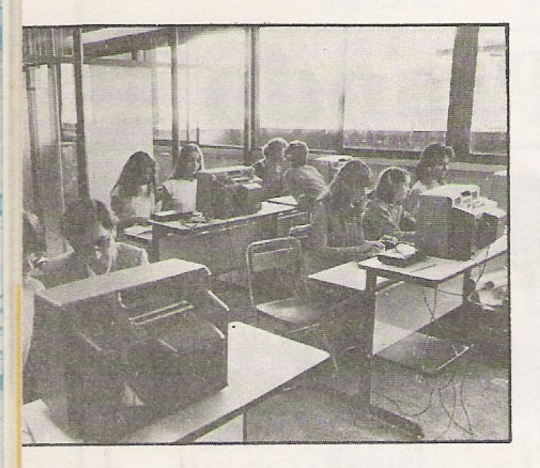
SOFTWARE
.UTILITARIOS
.LENGUAJES
.GESTION
.EDUCACION
.JUEGOS

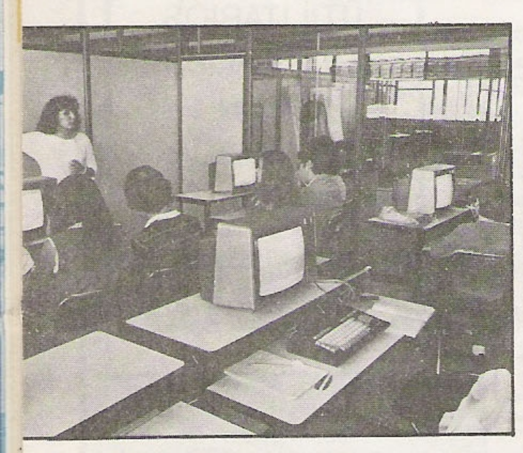
CURSOS
LIBROS

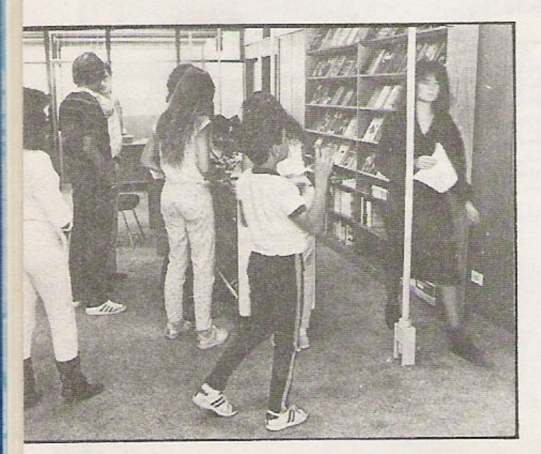
Talent MSX

CONOZCA LA DIFERENCIA...

Los socios del club de usuarios de MSX pueden desarrollar variadas actividades en su sede de Cabildo 2027, 1er. piso, Capital Federal:
Uso del computador con sus accesorios (impresora, robot, disketera, etc.) en modernas instalaciones dentro del horario de lunes a viernes de 8 a 22 y los sábados de 8 a 13.
En la biblioteca cuentan con libros específicos (los detallamos más adelante) los cuales se podrán fotocopiar y adquirir con un descuento especial.
Utilización de cualquiera de los títulos que forman banco de programas (utilitarios, compiladores, juegos, programas educativos, etc.)
Asesoramiento en el uso de los equipos, realización de programas etc.,
Cartelera de canjes de programas, revistas, y libros.
Alquiler de cartridge por períodos de 3 o 5 días.
Descuentos especiales para todos los cursos que se dicten en el Centro para el Desarrollo de la Inteligencia (CEDEI, ver cuadro).









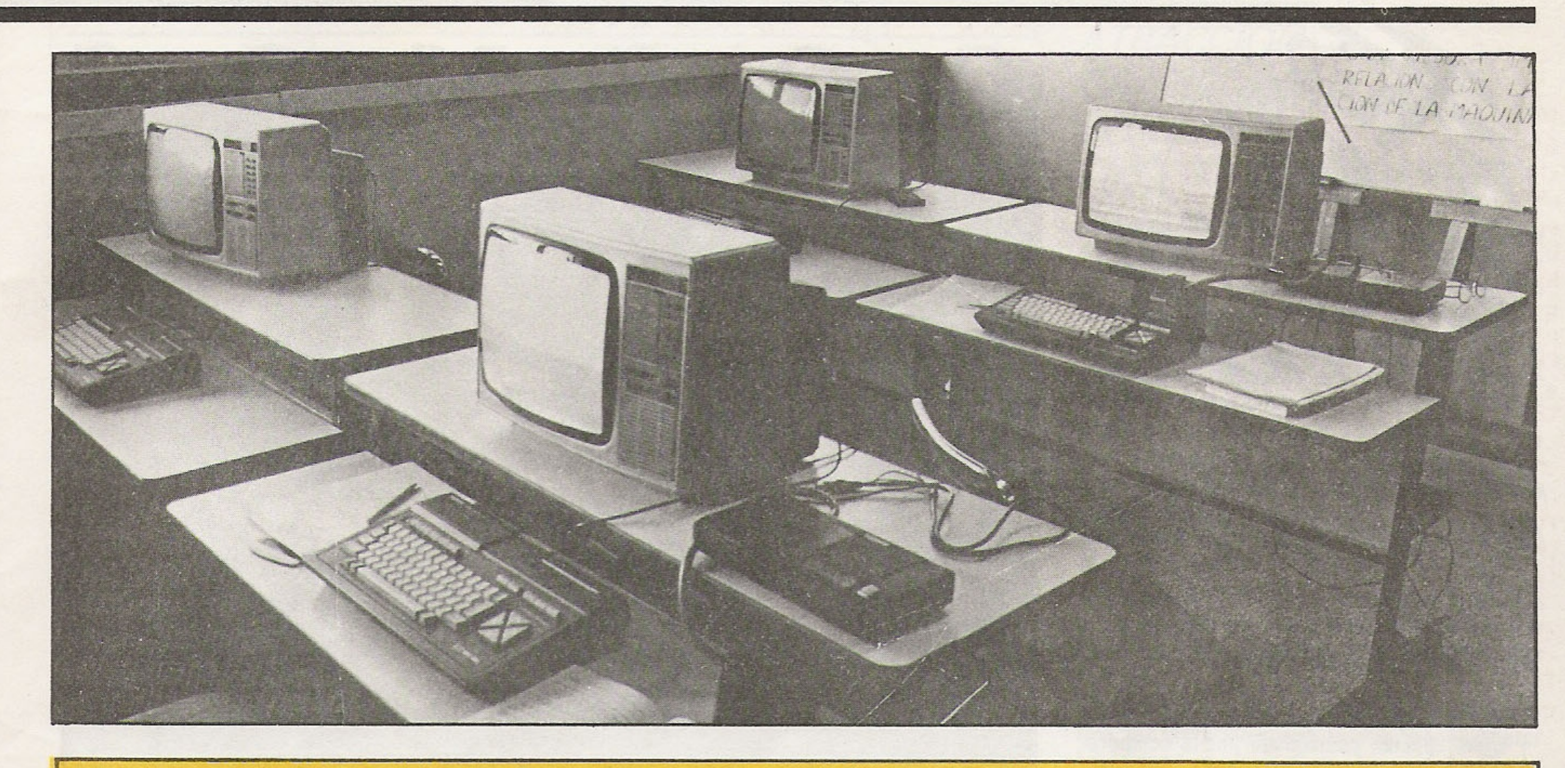

## BIBLIOGRAFIA DISPONIBLE

### GENERAL

| | |
|---|---|
| AGELOFF ROY | PROGRAMACION BASIC APLICADO |
| APPS VINCE | SELECCION DE PROGRAMAS PARA ORDENADORES |
| ASHTON TATE | MANUAL DEL USUARIO DE DBASE II |
| BELLIDO | BASIC PARA MAESTROS |
| BOSSUET | LA COMPUTADORA EN LA ESCUELA |
| CANOSA | EL SISTEMA UNIX Y SUS APLICACIONES |
| CHERRY TURKLE | EL SEGUNDO YO |
| CHORNOGUBSKY | LA NUEVA MICROCOMPUTADORA |
| CHRISTIE | ENCICLOPEDIA |
| DE ROSS | BASIC CURSO ACELERADO |
| DOMINGUEZ | PROGRAMACION EN BASIC |
| D'OPAZO | PROGRAMACION EN LOGO |
| FREEDMAN | GLOSARIO DE COMPUTACION |
| GRAVER | COBOL ESTRUCTURADO |
| LEVINE | INTRODUCCION A LA COMPUTACION |
| MANGADA | BASIC PARA NIÑOS |
| MANGADA | BASIC AVANZADO PARA NIÑOS |
| MARTIN | ORGANIZACION DE LAS BASES DE DATOS |
| MONTIEL | PRIMEROS PASOS EN LOGO |
| PHILIPAKIS | COBOL ESTRUCTURADO |
| REGGINI | ALAS PARA LA MENTE |
| SANDERS DONAL | INFORMATICA PRESENTE Y FUTURO |
| SAYMOUR | DESAFIO A LA MENTE |
| TENENBAUM | ESTRUCTURA DE DATOS EN PASCAL |
| WAITE | PROGRAMACION EN C |

### ESPECIFICA MSX

| | |
|---|---|
| DULLIN | MSX LENGUAJE DE MAQUINA |
| GUERREIRO | LOS SECRETOS DEL MSX |
| HOFFMAN | MSX GUIA DEL USUARIO |
| LACEY | MSX EL LIBRO GIGANTE DE LOS JUEGOS |
| LEPAPE | PROGRAMACION DEL Z80 CON ENSAMBLADOR |
| LOWY FRUTOS | MSX PROGRAMACION GRAFICOS, COLORES Y MUSICA |
| LUERS | MSX PROGRAMAS Y UTILIDADES |
| LUERS | MSX GRAFICOS Y SONIDOS |
| NICHOLS | MICROPROCESADOR Z80 PROGRAMACION |
| PEARCE | MSX PROGRAMACION BASICA |
| POITIS | INTRODUCCION AL MSX |
| PRITCHARD | DESCUBRE TU MSX |
| PRITCHARD | LENGUAJE DE MAQUINA MSX |
| SATO | MSX GUIA DEL PROGRAMADOR - MANUAL DE REFERENCIA |
| SINCLAIR | PROGRAMANDO CON EL MSX BASIC |
| VOSS | MSX EL MANUAL ESCOLAR |
| WEBB | MSX CODIGO DE MAQUINA |
| ZAKS | PROGRAMACION DEL Z80 |
| | MSX LOGO |
| | MANUAL DE USO TALENT MSX |
| | MSX TECHNICAL DATA BOOK |
| | REVISTA MSX CLUB |
| | REVISTA MSX ORDENADORES |
| | REVISTA MSX MAGAZINE |




| CURSO | * REQUISITOS | HORARIOS |
|---|---|---|
| BASICO DE USO | | LU., MI. Y VI. DE 10:30 A 12:30<br>LU., MI. Y VI. DE 17:00 A 19:00<br>SA. DE 09:00 A 12:00 |
| INTRODUCCION A LA INFORMATICA I | | |
| INTRODUCCION A LA INFORMATICA II | | LU., Y MI. VI. DE 08:30 A 10:30<br>LU., Y MI. DE 18:00 A 20:00<br>MA. Y JU. DE 20:00 A 22:00 |
| DIAGRAMACION ESTRUCTURADA | | LU., Y MI. DE 08:30 A 10:30<br>MI., Y VI. DE 20:00 A 22:00 |
| BASIC NIÑOS I | | JU DE 09:00 A 11:00<br>VI. DE 18:00 A 20:00<br>SA. DE 09:00 A 11:00 |
| BASIC NIÑOS II | BASIC NIÑOS I | |
| BASIC JOVENES I | | MI. Y VI. DE 09:00 A 11:00<br>MA. Y JU. DE 18:00 A 20:00 |
| BASIC JOVENES II | BASIC JOVENES I | |
| BASIC ADULTOS I | INTRODUCCION A LA INFORMATICA | LU. Y JU. DE 19:00 A 21:00<br>MA. Y VI. DE 20:00 A 22:00 |
| BASIC ADULTOS II | BASIC ADULTOS I | |
| LOGO NIÑOS | | MA. DE 09:00 A 11:00 |
| LOGO PROFESIONALES | INTRODUCCION A LA INFORMATICA | |
| COBOL | INTRODUCCION A LA INFORMATICA<br>DIAGRAMACION ESTRUCTURADA | |
| PASCAL | INTRODUCCION A LA INFORMATICA<br>DIAGRAMACION ESTRUCTURADA | |
| ESTRUCTURA DE DATOS EN PASCAL | PASCAL | |
| INTRODUCCION AL ASSEMBLER | | |
| ASSEMBLER I | BASIC I | MA. Y JU. DE 18:00 A 20:00 |
| ASSEMBLER II | ASSEMBLER I | |
| DBASE II | INTRODUCCION A LA INFORMATICA | LU. Y JU. DE 20:00 A 22:00 |
| MSX DOS | INTRODUCCION A LA INFORMATICA | |
| WORD PROCESSING | | |
| MULTIPLAN | | |
| SPRITES Y SONIDOS (BASIC) | BASIC I Y II | |
| ARCHIVOS (BASIC) | BASIC I Y BASIC II | |

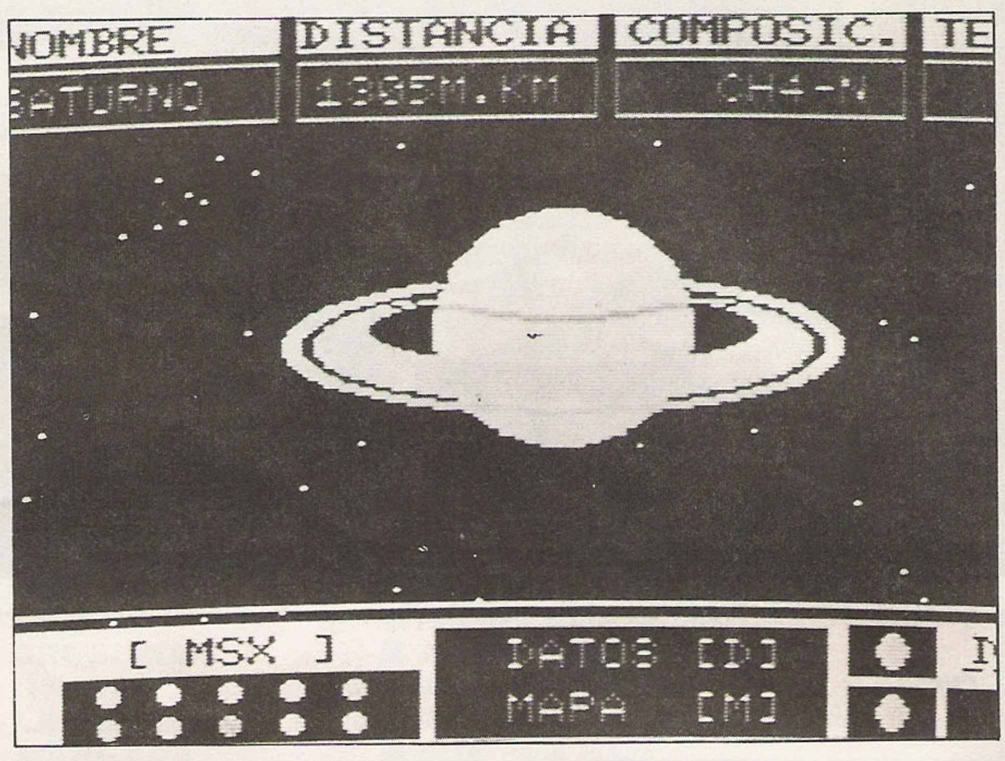
* HABER RECIBIDO EL CURSO QUE SE INDICA O POSEER CONOCIMIENTOS EQUIVALENTES

TODOS LOS CURSOS INDICADOS CON SU CORRESPONDIENTE HORARIO COMIENZAN A PARTIR DEL 5 DE MAYO

Raiting Soft

# COSMOS

CREATIVIDAD: **9**
PRESENTACION: **9**
ATRACCION: **8**
GRAFICOS: **8**
SONIDO: **7**
TIPO: **EDUCATIVO**




**Es un viaje a bordo de un cohete espacial, comandado por la computadora a través de nuestra vía láctea. Desde la cuenta regresiva el despegue y hasta el viaje, son simulaciones alucinantes.**

**Todo el sistema solar aparece en la pantalla. Con nuestra nave podemos aterrizar en cualquier planeta o inclusive en el cinturón de asteroides entre Marte y Júpiter.**

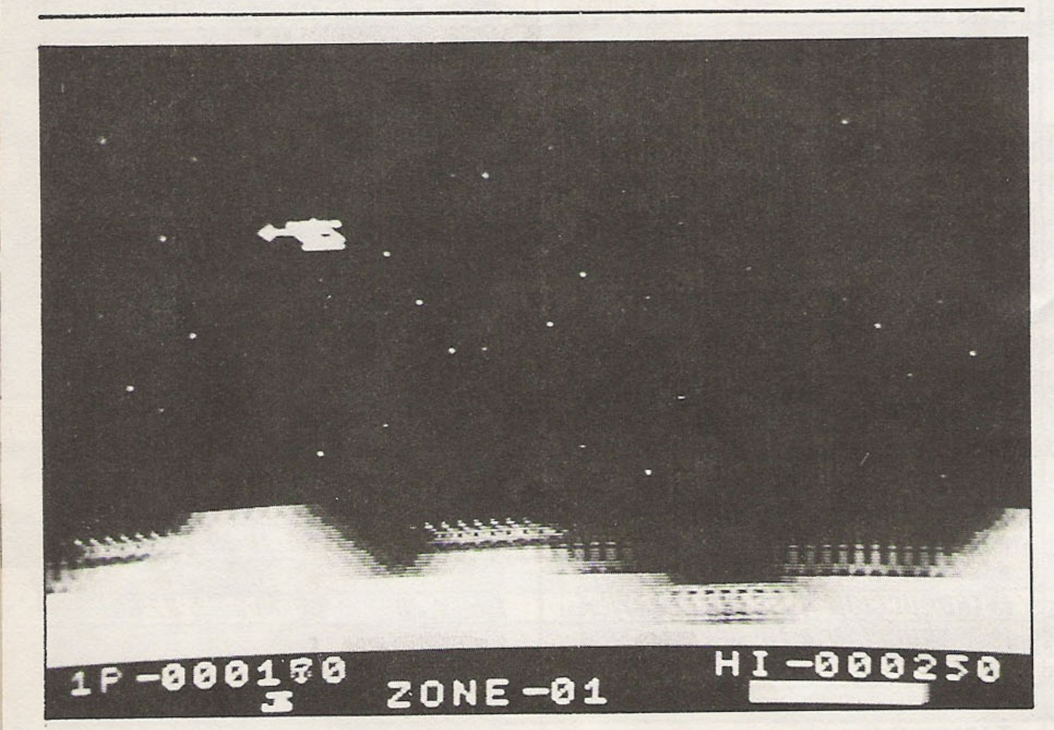
**El sistema de percepción computarizada de nuestro taxi espacial, nos informará entre otras cosas sobre el nombre, ubicación, composición, distancia y temperatura del planeta que visitamos.**

**Respondiendo las preguntas que aparecen en la parte inferior del tablero, obtendremos datos más precisos como albedo, revolución sideral y sinódica del astro. Algunos términos nos pueden parecer raros, pero en la memoria de nuestra nave se encuentran las explicaciones para cada dato y esta información es accesible cuando nos pregunte por "términos".**

**En la información de los planetas, están incorporados los datos sobre los satélites. Por ejemplo, Saturno es dibujado con sus anillos. Además de la información sobre este planeta, podemos saber quiénes (y cuándo) descubrieron sus diez satélites.**

**Los gráficos son buenos y la información es completa, pero faltarían algunos detalles musicales para optimizar este programa que no dejará de sorprender a ningún astronauta que se atreva a incorporarse a la tripulación de COSMOS.**

# SUPER COBRA




CREATIVIDAD: **9**
PRESENTACION: **8**
ATRACCION: **9**
GRAFICOS: **9**
SONIDO: **8**
TIPO: **ENTRETENIMIENTO**

**Es un juego en el cual piloteamos un helicóptero con la misión de recoger una caja y regresar a nuestra base. En el camino seremos atacados por misiles enemigos, que serán arrojados desde distintos puntos y no cesarán hasta destruirnos.**

**El helicóptero está equipado con dos sistemas de defensa. Al disparar con la barra de espacio o con el botón del joystick, lanzará proyectiles hacia abajo y otros en dirección horizontal al mismo tiempo.**

**Nuestro vehículo volador consume bastante combustible y un sonido especial nos avisará cuando las reservas se estén agotando. La manera de recargar gasolina es destruyendo con bombas los tanques que tengan una letra "F". Si no podemos reabastecernos de combustible, el helicóptero caerá.**

**A medida que avancemos sobre el terreno del enemigo, aumentan las dificultades y pasamos de nivel.**

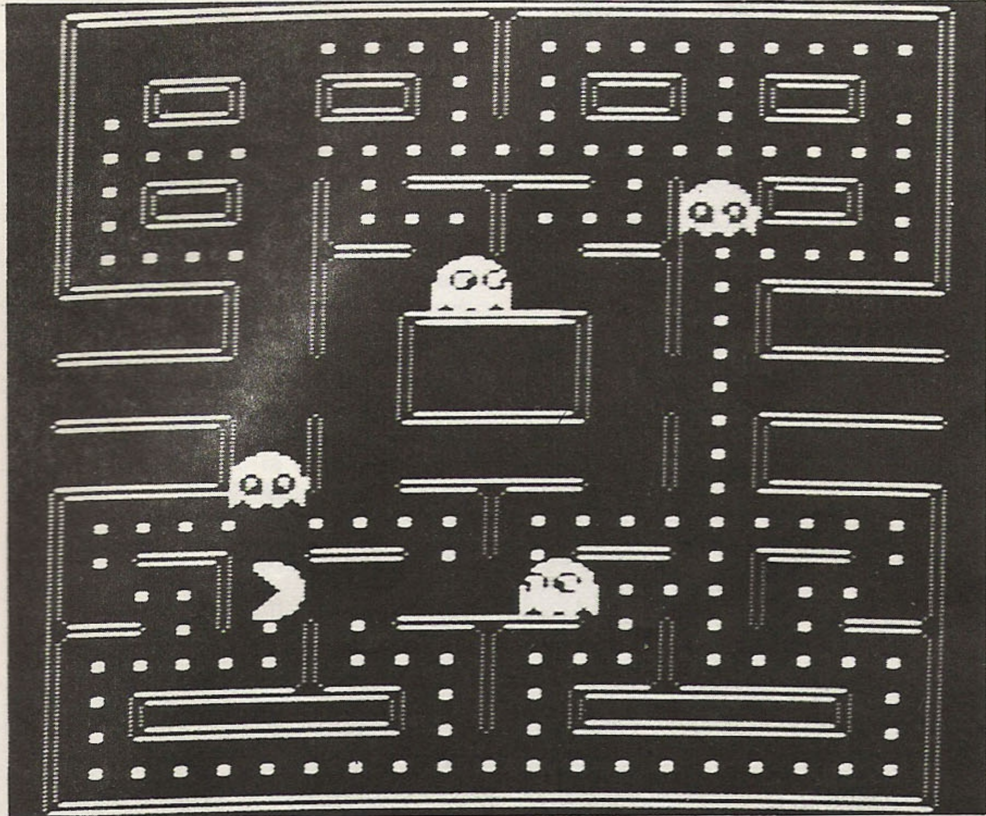
**Los gráficos que utiliza este programa están muy bien logrados, acompañados por una combinación de colores agradables.**

**El helicóptero puede moverse en las cuatro direcciones para poder esquivar los obstáculos y misiles agresores.**

**Es un juego de mucha acción que nos mantendrá con los sentidos activados durante todo el programa. Pero para amortiguar nuestra tensión, el helicóptero parece disponer de un equipo de música que hará sonar, durante nuestra misión, una agradable melodía.**

# P A C  M A N

CREATIVIDAD: **8**
PRESENTACION: **7**
ATRACCION: **8**
GRAFICOS: **9**
SONIDO: **9**
TIPO: **ENTRETENIMIENTO**



Es una versión muy buena del clásico juego que nos introdujo a muchos en el mundo del entretenimiento con los computadores.

Este soft es un laberinto y cada camino contiene puntos amarillos que deben ser devorados por nuestro monstruito antes que los fantasmas nos atrapen.

Para refrescarnos, aparecerán frutas y si no somos despreciativos y las comemos, seremos recompensados con puntaje.

Cada nivel tiene más dificultades y una fruta distinta.

Gráficamente, los personajes de este juego tienen una definición favorable, compuesta por varios diseños según sea el movimiento que realizan.

Si nuestro PAC MAN está haciendo la digestión, los fantasmas tomarán otra imagen y color hasta que termine de digerir los puntos gruesos que se hayan distribuido en los cuatro vértices.

Este pasatiempo, tiene una agradable melodía al comenzar y durante todo el juego, produce distintos ruidos que nos indicarán cuándo podemos comer a los fantasmas o huir de ellos. Es un juego para no despegar los dedos del teclado y para pensar solamente en evadirse de fantasmas comiendo la mayor cantidad de puntitos y frutas.

# FLIGHT PATH 737

Si alguna vez soñó con maniobrar un avión, ésta es una buena oportunidad. No es fácil manejar una máquina de este tipo y este simulador de vuelo lo demostrará.

Deberemos pilotear una nave comercial y para desafiar nuestros conocimientos como pilotos, contamos con varias opciones.

Podremos despegar en una pista de 3 millas con montañas enfrente de 5.000 pies de altura. Pero si esto no es dificultoso y conseguimos aterrizar con todos los pasajeros a salvo, tratemos de volver a intentarlo, pero con montañas de 6.000 pies y un incendio en el motor.

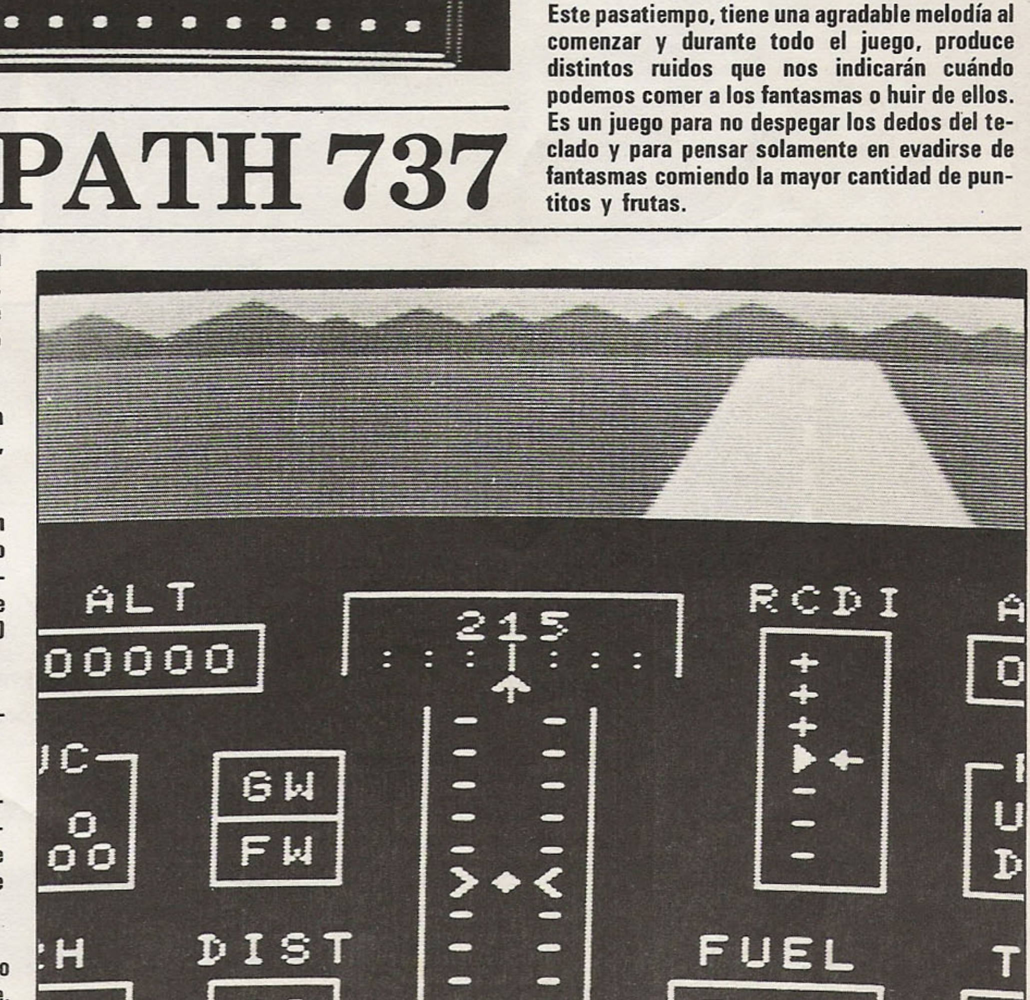
No importa que algún pasajero se haya mareado por nuestras maniobras.

Las dificultades pueden aumentar hasta convertirnos en una especie de magos para despegar en una pista de 1,5 millas con montañas de 9.200 pies y soportando vientos de costado e incendios.

El juego viene acompañado por un pequeño manual de instrucciones sobre el aterrizaje, despegue y vuelo de avión.

La presentación del juego tiene un conjunto de sonidos diferentes combinados con una armonía de estilo francés.

No se desaliente si las primeras veces no logra maniobrar su máquina, lo importante es insistir. Pero trate de viajar sin pasajeros hasta que le haya tomado la mano a los comandos.




CREATIVIDAD: **9**
PRESENTACION: **7**
ATRACCION: **9**
GRAFICOS: **8**
SONIDO: **8**
TIPO: **SIMULACION**

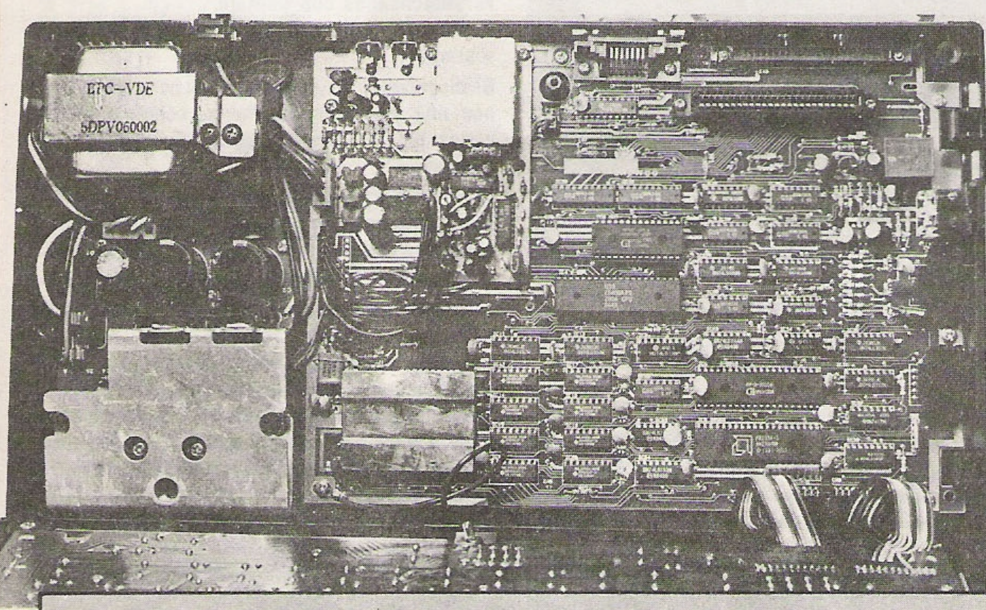
# RADIOGRAFIA DE LA TALENT MSX

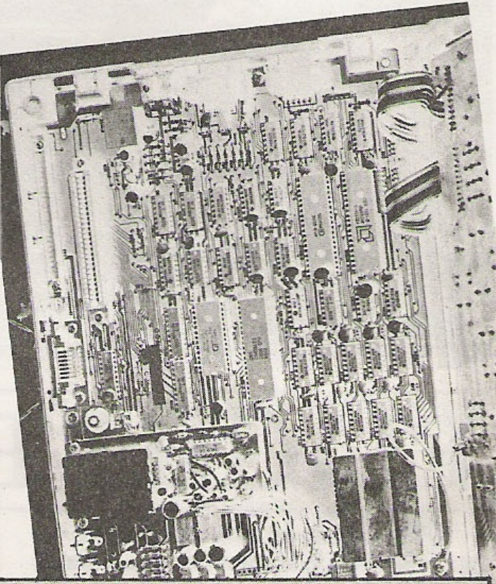
## Vamos a desnudar la computadora. Veremos algunos de los circuitos básicos que componen a nuestra Talent MSX.

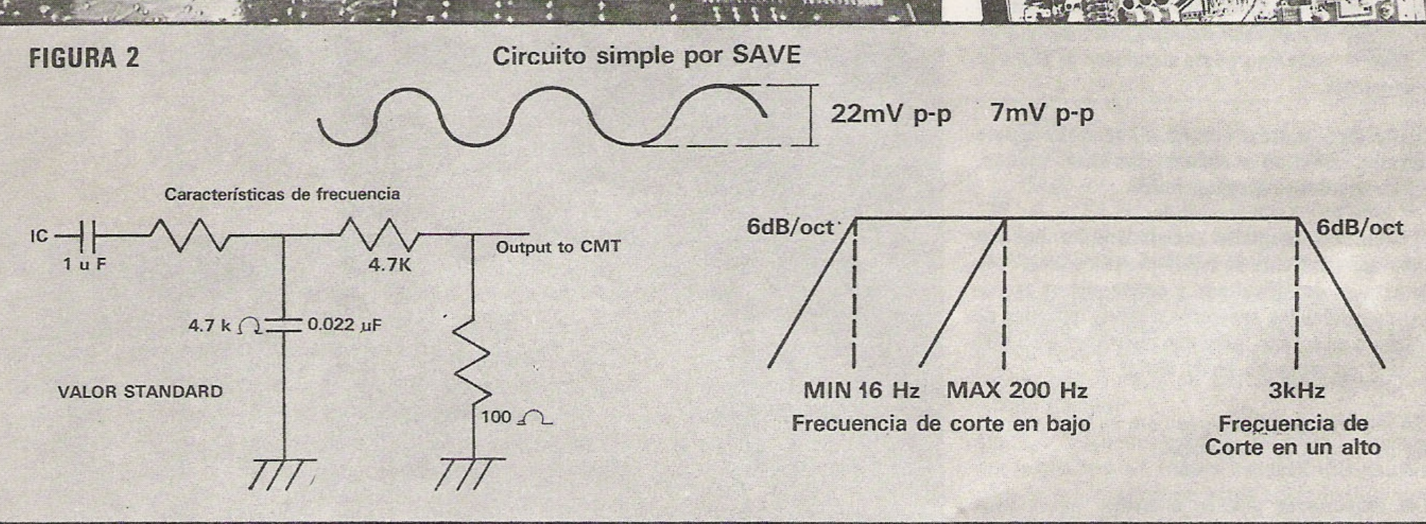
Esta nota será de interés para todos aquellos que deseen conocer las características internas de la máquina, para el desarrollo de futuros proyectos de hardware. Será necesario, para comprenderla, tener conocimientos básicos de assembler y electrónica digital.

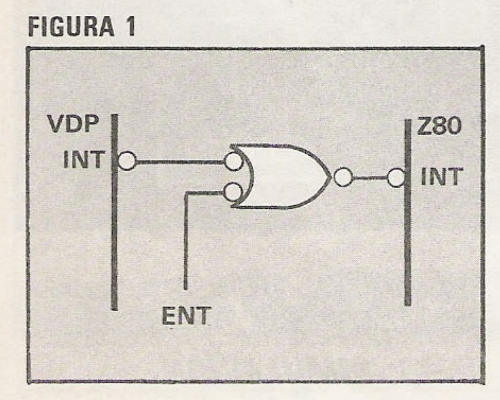
### Interrupciones

En lo que a hard se refiere, el Z80 posee dos interrupciones:








| PIN NO. | NOMBRE DE LA SEÑAL | DIRECCION | CONECTORES |
|---|---|---|---|
| 1 | ARRIBA | ENTRADA | (1) (2) (3) (4) (5) (6) (7) (8) (9) |
| 2 | ABAJO | ENTRADDA | |
| 3 | IZQUIERDA | ENTRADA | |
| 4 | DERECHA | ENTRADA | |
| 5 | +5V | | |
| 6 | DISPARADOR1 | ENTRADA/SALIDA | |
| 7 | DISPARADOR2 | SALIDA | |
| 8 | SALIDA | SALIDA | |
| 9 | MASA | | |

1) NMI (Nonmaskable Interrupt) Esta interrupción no es usada por el sistema MSX pues provoca un llamado a la posición de memoria H0066 que utiliza el MSX-DOS para datos.
2) INT (Interrupt Request) El sistema MSX usa esta interrupción por software con el VDP. Estas interrupciones se realizan con una frecuencia de 50 Hz en el sistema PAL/SECAM.
Ver figura 1

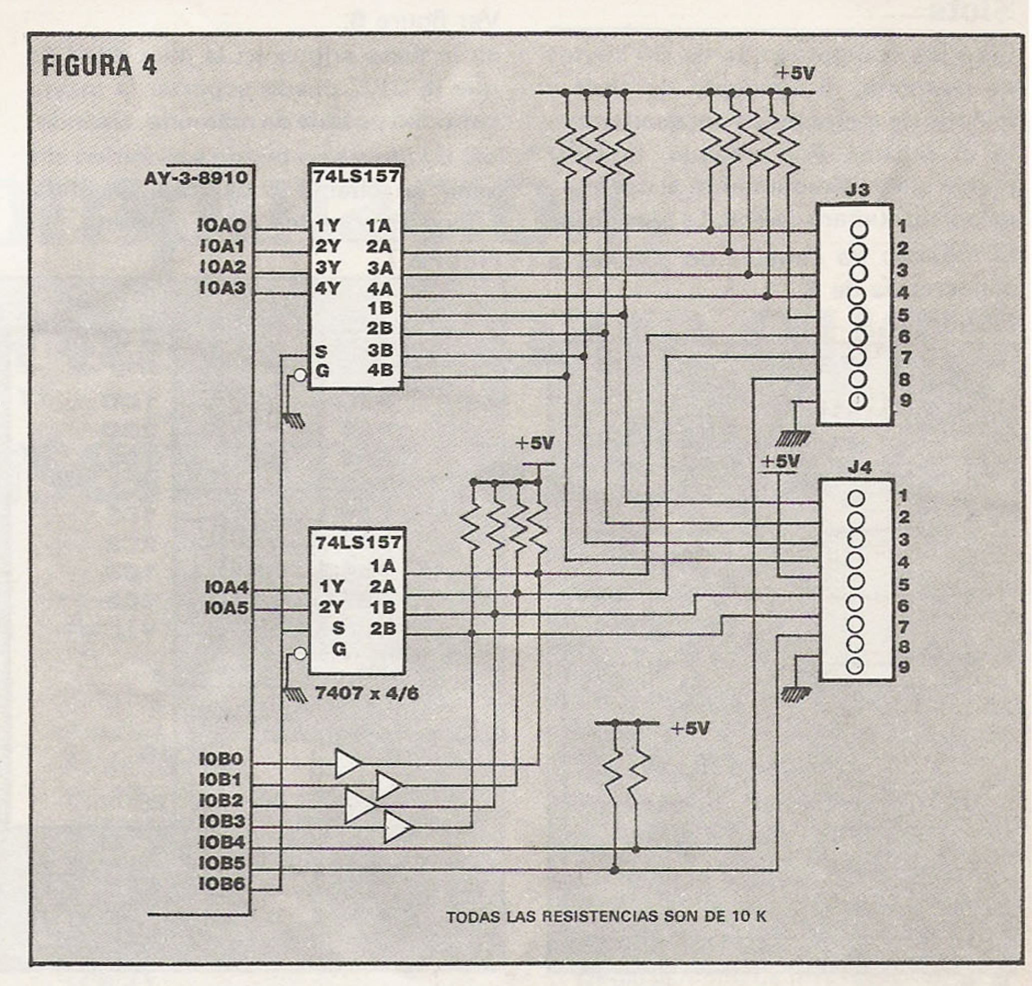
## Nivel de grabación

Las constantes en el circuito de salida correspondiente a la instrucción SAVE, pueden ser ajustadas de la siguiente forma:
Nivel de salida: -45 dBm /- 5dBm (0 dBm 0.775 V)
Nivel de tensión de salida: 22mVp-p a 7 mVp-p con una señal de 1200 Hz. (Ver figura dos).
El capacitor de corte está simplmente para proteger al integrado de la MSX. La capacitancia puede fluctuar entre .1 y 2.2 uf. Su valor se ajusta al límite de la frecuencia de corte (entre 16 y 200 Hz), si la impedancia de salida del integrado es demasiado alta.

## Pórticos de entrada/salida (Joystick)

Integrado: AY-3-8910 o compatible
Entrada/Salida: 4 bits de entrada, 1 bit de salida, 2 bits bidereccionales por pórtico.
Lógica: TTL
Nivel: Se activa con un alto.
Conector: 9 pines AMP o compatible.
Lista de pines: Ver figura 3.
Corriente: 50 mA
Circuito esquemático: Ver figura 4.
Interpretación: Ver figura 5.

FIGURA 4



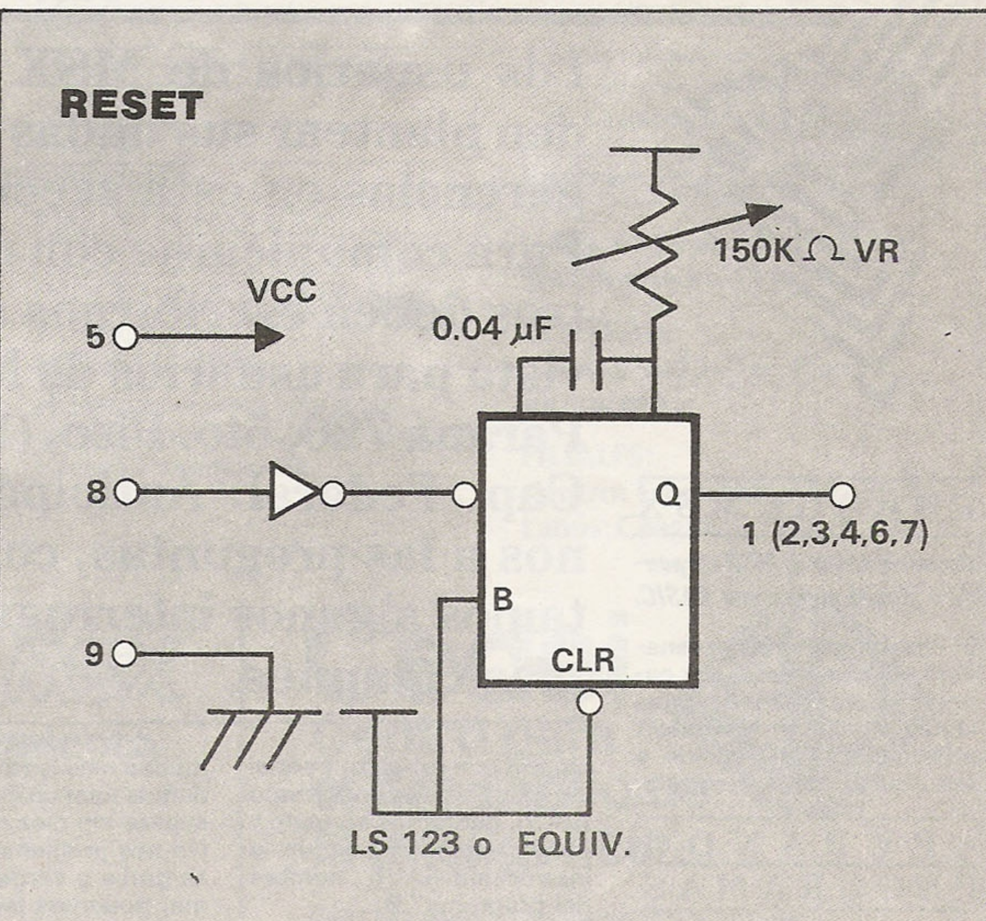
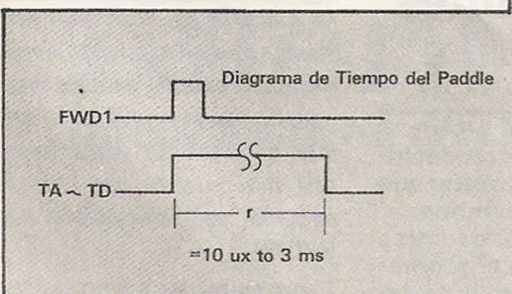




FIGURA 6




## Paddles

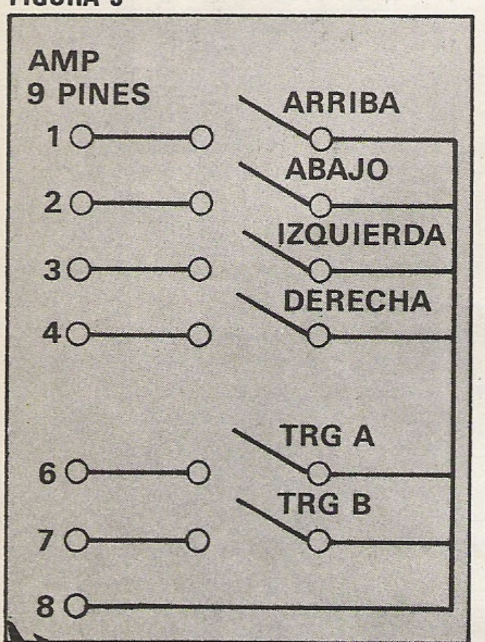
Un pulso de disparo es mandado al pin 8 del pórtico de E/S cada vez que se llama a la función PDL. El circuito asociado al paddle dispara a un multivibrador monoestable con el pulso al que raciamos referencia.
Lo que retorna al pórtico es un pulso cuya longitud representa el nivel de volúmen.

## Slots

Para las computadoras de 64 kbytes de memoria, el concepto de slots y bancos de memoria es necesariamente el mismo. Por ejemplo, la CPU puede elegir directamente al cartrige por su numero de slot. La idea de los slots tiene orígen en la necesidad de que la CPU pueda soportar la mayor cantidad posible de memoria. Usándolos, un programa puede ser corrido sin tener en cuenta el número de slots físicos disponibles en la máquina.

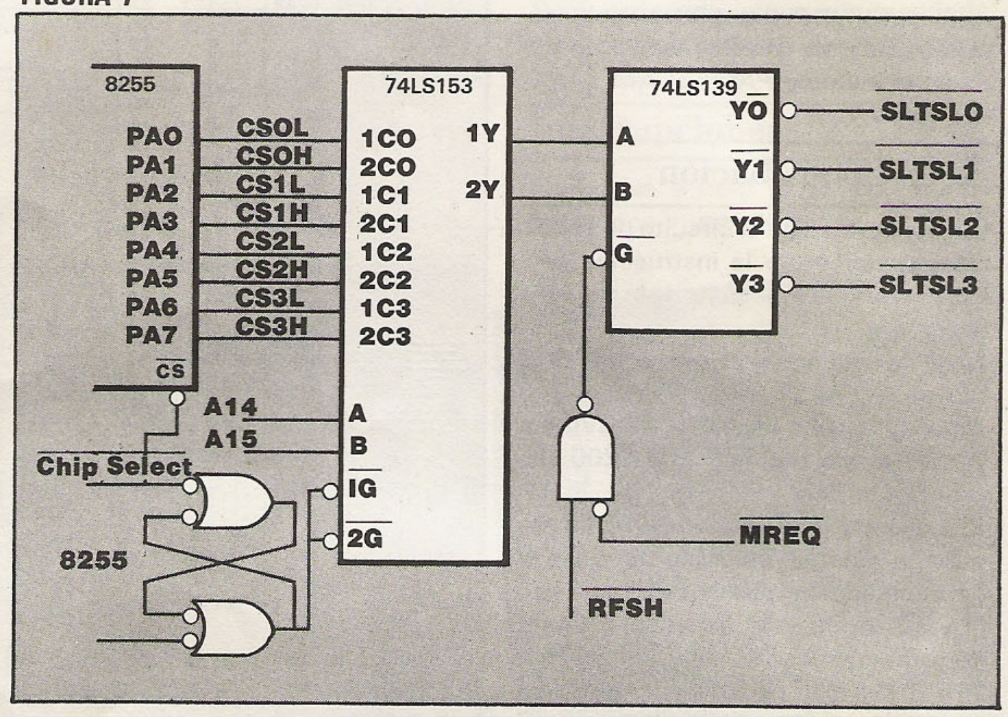
El máximo de canales de paddles a conectar es de 6.

**FIGURA 5**




Ver figura 6.

**FIGURA 7**




El diagrama del circuito que se utiliza para lograrlos es el que se ve en la figura 7.

Esperamos que les sea de utilidad, y que atomice los bloques creativos de las mentes impacientes por lograr nuevos desarrollos.




**Los usuarios de MSX pueden plantear sus dudas y sugerencias en esta sección. Para comunicarse con nosotros deben escribirnos a Revista para usuarios de MSX, Paraná 720, 5to. Piso, (1017) Cap. Federal. Anticipándonos a las preguntas, contestamos algunos interrogantes más comunes.**

### PORQUE MSX

***Desearía saber si MSX responde a alguna sigla como BASIC.***

El nombre de estos ordenadores no fue puesto por capricho o casualidad. Estas siglas significan MicroSoft eXtended BASIC debido a que creó el nuevo intérprete.

### APURANDO PROGRAMAS

***¿Es posible cargar un programa y ejecutarse inmediatamente?***

Los programas generalmente son grabados con CSAVE y son guardados en formato binario comprimido. Pero si los guardamos con formato ASCII, podremos cargarlo y hacerlo ejecutar usando la instrucción SAVE "nombre del programa",R.

Pero sería iluso pensar que no tendremos alguna desventaja si grabamos y cargamos de esta manera. Pues bien, el tiempo que tardamos en realizar estas dos operaciones en ASCII es mucho más lento; recomendamos usar un cassette que supere los diez minutos.

No hay problema: mientras se graba y carga el programa, podemos leer la revista.

### BLOQUEO

***En una cinta tengo guardados varios programas y después de FOUND y del OK no vuelve a aparecer el cursor, aunque deje correr la cinta.***

Si la cinta no carga bien, suele bloquear la máquina. Puede suceder por no haber seguido al pie de la letra las instrucciones sobre el cargado de programas y sería conveniente también ajustar el cabezal del grabador. No es aconsejable grabar los programas varias veces en un mismo sector de cinta. Tratemos de no regrabar en el mismo lugar más de cinco o seis veces.

### CENTRADO DE PANTALLA

***La pantalla se ve ligeramente descentrada hacia la izquierda y cuando escribo en este margen, no puedo visualizar el primer caracter. La única forma que me permite ver toda la pantalla es achicándola con WIDTH.***

El problema no creemos que sea del computador. Habría que ajustar y centrar la pantalla con los mandos que suelen tener los monitores y televisores. Pero aconsejamos que llamen a alguien que sepa hacerlo.

# Computación, una oportunidad para que todos enseñen y aprendan.

## Un lugar para

desarrollar el pensamiento.

descubrir una vocación.

manejar lenguajes de computación.

comprender los múltiples usos de un computador.

capacitar y perfeccionar al docente.

incorporar los avances tecnológicos.

que el profesional domine el uso de nuevas herramientas.

que los padres se reencuentren con sus hijos.

"No se trata solamente de adquirir en forma puntual conocimientos definitivos, sino prepararse a elaborar a lo largo de toda la vida, un saber en constante evolución y de aprender a ser."

UNESCO

## Actividades '86

Para Niños, Adolescentes, Adultos, Docentes, Profesionales y Establecimientos educativos.

INTRODUCCION A MICROCOMPUTADORES

DIAGRAMACION ESTRUCTURADA

LOGO

BASIC

COLOR - SPRITE - SONIDO

COBOL

PASCAL

ASSEMBLER

MS - DOS Y MSX - DOS

D BASE II - MULTIPLAN

PROCESADOR DE LA PALABRA

INSTALACION DE LABORATORIOS

en Establecimientos educativos con formación de multiplicadores y apoyo a la comunidad.

## Cómo?

- Taller en grupos de 12 a 15 personas.
- Clases de 2 horas diarias.
- 2 ó 3 alumnos por equipo.
- Equipos disponibles para prácticas adicionales en horarios libres.
- Becas rentadas en el Departamento de investigación y desarrollo de Talent MSX.
- Becas rentadas para docentes en Laboratorios de Establecimientos Educativos.

## Informes, Inscripción y Cursos

**Lunes a Viernes de 8 a 22 hs.**
**Sábados de 8 a 13 hs.**

**CENTRAL:**
**Cabildo 2027 - 1er. Piso y Juramento**

**FILIALES:**
**Centro: Esmeralda 320 - 5º P.**
**Lanús: Caaguazú 2186 - L. Este**

**Talent MSX**
**Inteligencia en crecimiento.**

**Cedei**
**Centro para el desarrollo de la inteligencia.**

Descubramos y construyamos juntos los caminos que nos permitirán el uso inteligente de los productos de la creatividad humana.

diálogo - *D.P.*


Títulos no oficiales

# A la computadora personal

# Talent MSX

# nada le es imposible

dialogo - D.P.

MSX es marca registrada de MICROSOFT CORPORATION

Porque gracias a la norma internacional MSX, la TALENT MSX trasciende sus propios límites.
Hasta ahora, cuando usted compraba una computadora personal de cualquier marca, quedaba automáticamente desconectado del resto del mundo de la computación. Porque los distintos equipos y sistemas no eran compatibles entre sí.
Hasta que dos grandes empresas de informática, la Microsoft Corp. de EE.UU. y la ASCII del Japón se pusieron de acuerdo para crear una norma standard: la MSX. Que se expandió también rápidamente en Europa. Y que hoy TALENT presenta por primera vez en la Argentina.
Mientras que la mayoría de las computadoras de su tipo que se ofrecen en el mercado nacional, han sido discontinuadas por obsoletas en sus lugares de origen, TALENT MSX tiene casi ilimitadas posibilidades de desarrollo. Porque la norma MSX es en todo el mundo inteligencia en crecimiento.
La TALENT MSX pone a su disposición un mundo de software para elegir. Y con la incorporación de todos sus periféricos llega a ser una auténtica computadora profesional.

## UTIL

Su poderoso sistema operativo MSX permite el acceso a todo tipo de procesamiento de datos:

- Planillas de calculo.
- Procesadores de palabra.
- Graficos de negocios.
- Bases de datos (d Base II, etc.)
- Contabilidad general, sueldos, y jornales, costos, etc., desarrollados bajo CP/M en Basic, Cobol, Pascal o C.

Con la posibilidad de conexion a linea telefonica permite la transferencia y consulta de datos entre computadoras personales, profesionales o bancos de datos.
La grabacion de archivos es en formato MS-DOS, haciendola compatible con las computadoras profesionales.

## DIDACTICA

Dispone de tres lenguajes para la enseñanza de computacion: LOGO como lenguaje de induccion para los mas chicos. Lenguaje de Programacion en castellano, para todos los que quieran aprender a programar sin conocimientos previos. Y Basic MSX como lenguaje profesional. Mas una amplia variedad de perifericos como el Mouse, Lapiz Optico, Tableta grafica, Track-ball, etc.

## DIVERTIDA

La mas genial para Video-Juegos. Por la amplisima biblioteca de programas –**todos nuevos**– de la norma MSX en el mundo. Y ademas, el Basic MSX permite al usuario generar sus propios juegos con un manejo tan simple, como solo TALENT MSX puede ofrecer.

## CARACTERISTICAS TECNICAS

- Memoria principal 64 KB ampliable hasta 576 KB.
- Memoria de video: 16 KB RAM.
- ROM incorporada de 32 KB con el MSX-Basic de Microsoft.
- Graficos completos, hasta 32 sprites y 16 colores simultaneos.
- Generador de sonido de 3 voces y 8 octavas.
- Conexion para cualquier grabador.
- Interfaz para salida impresora paralela.
- Conectores para cartuchos y expansiones.
- Fuente para 220 V y modulador PAL-N incorporado.

**DISTRIBUIDORES AUTORIZADOS: CAPITAL FEDERAL:** AMATRIX, Bolívar 173 - ARGECINT, Av. de Mayo 1402 - BAIDAT COMPUTACION, Juramento 2349 - COMPUPRANDO, Av. de Mayo 965 - COMPUSHOP, Córdoba 1464 - COMPUTIQUE, Córdoba 1111, E. P. - COMPUTRONIC, Viamonte 2096 - CP67 CLUB, Florida 683, L. 18 - DALTON COMPUTACION, Cabildo 2283 - ELAB, Cabildo 730 - MICROSTAR, Callao 462 - Q.S.P., Bartolomé Mitre 864 - SERVICIOS EN INFORMATICA, Paraná 164 - DISTRIBUIDORA CONCALES, Tucumán 1458 - MICROMATICA, Av. Pueyrredón 1135 - **ACASSUSO:** MICROSTAR ACASSUSO, Eduardo Costa 892 - **AVELLANEDA:** ARGOS, Av. Mitre 1755 - **BOULOGNE:** COMPUTIQUE CARREFOUR, Bernardo de Irigoyen 2647 - **CASTELAR:** HOT BIT COMPUTACION, Carlos Casares 997 - **LANUS:** COMPUTACION LANUS, Caaguazú 2186 - **LOMAS DE ZAMORA:** ARGESIS COMPUTACION, Av. Meeks 269 - **MARTINEZ:** VIDEO BYTE, Hipólito Yrigoyen 32 - **RAMOS MEJIA:** MANIAC COMPUTACION, Rivadavia 13734 - **SAN ISIDRO:** FERNANDO CORATELLA, Cosme Beccar 249 - **VICENTE LOPEZ:** SERVICIOS EN INFORMATICA, Av. del Libertador 882 - **BAHIA BLANCA:** SERCOM, Donado 327 - SUMASUR, Alsina 236 - **LA PLATA:** CADEMA, Calle 7 Nº 1240 - CERO-UNO INFORMATICA, Calle 48 Nº 529 - **MAR DEL PLATA:** FAST, Catamarca 1755 - **NECOCHEA:** CAFAL, Calle 57 Nº 2920 - SERCOM, Calle 57 Nº 2216 - **TRENQUE LAUQUEN:** COMPUQUEN, Villegas 231 - **CORDOBA:** AUTODATA, Pasaje Santa Catalina 27 - TECSIEM, Santa Rosa 715 - **ROSARIO:** 2001 COMPUTACION, Santa Fe 1468 - MINICOMP, Maipú 862 - SISOR, Urquiza 1062 - **SANTA FE:** ARGECINT, P. San Martín 2433, L. 36 - SISOR, Rivadavia 1062 - INFORMATICA, San Gerónimo 2721/25 - **VILLA MARIA:** JUAN CARLOS TRENTO, 9 de Julio 80 - **MENDOZA:** INTERFACE, Sarmiento 98 - BIT & BYTE, 9 de Julio 1030 - **COMODORO RIVADAVIA:** COMPUSER, 25 de Mayo 827 - **GENERAL ROCA:** DISTRIBUIDORA VECCHI, 25 de Mayo 762 - **LA PAMPA:** MARINELLI, Pellegrini 155 - **NEUQUEN:** MEGA, Perito Moreno 383 - EDISA, Roca esq. Fotheringham - **RIO GRANDE:** INFORMATICA M & B, Perito Moreno 290 - **SAN CARLOS DE BARILOCHE:** L. ROBLEDO & ASOCIADOS, Elfein 13, Piso 1º - **TRELEW:** SISTENOVA, Sarmiento 456 -